数 值 分 析

第4版

李庆扬　王能超　易大义　编

清华大学出版社
施普林格出版社

(京)新登字158号

内 容 提 要

本书是为理工科大学各专业普遍开设的“数值分析”课程编写的教材.其内容包括插值与逼近,数值微分与数值积分,非线性方程与线性方程组的数值解法,矩阵的特征值与特征向量计算,常微分方程数值解法.每章附有习题并在书末有部分答案,书末还附有计算实习题和并行算法简介.全书阐述严谨,脉络分明,深入浅出,便于教学.

本书也可作为理工科大学各专业研究生学位课程的教材,并可供从事科学计算的科技工作者参考.

书　名:数值分析(第4版)
作　者:李庆扬　王能超　易大义　编
出版者:清华大学出版社　施普林格出版社
(北京清华大学学研大厦,邮编100084)
http://www.tup.tsinghua.edu.cn
印刷者:北京振华印刷厂
发行者:新华书店总店北京发行所
开　本:850×1168 1/32　印张:13.125　字数:328千字
版　次:2001年8月第4版　2002年4月第2次印刷
书　号:ISBN 7-302-04561-5/O・259
印　数:5001～10000
定　价:16.00元

第四版前言

本书由华中理工大学出版社出版至今已 20 年，重新修订的第三版也已 15 年了，印数已近 20 万册，1988 年获国家教委优秀教材二等奖，表明本教材在国内是受欢迎的，仍有存在的价值. 为使本书适应新世纪的要求，我们认为对本书重新进行修改是完全必要的. 这次修改除保留本书原有风格和基本内容外，修改的原则和内容有以下几点：

(1) 随着计算机技术的发展和普及，数值分析的原理与方法在各学科中的应用越来越多. 因此，我们将原来主要面向应用数学专业扩大为面向理工科大学中对数学要求较高的专业的本科生，同时也兼顾到一些院校为各专业研究生开设的"数值分析"学位课程.

(2) 由于科学及计算机的发展，计算机算法语言的多样化及数学软件的普及，要求"数值分析"课程更强调算法原理及理论分析，而对具体算法及编程已有现成数学软件，如 Matlab 等，方便了读者使用. 因此，我们对某些算法做了精简，另外也删去了一些较少使用的算法，增加一些实际应用中较重要的内容，如帕德逼近，解线性方程组的 QR 方法及超定方程组最小二乘解，非线性方程组求解的牛顿法，解刚性常微分方程的基本概念等. 考虑到很多高校配备了大型多处理机，具备了进行并行计算的条件，故增加了"并行算法及其基本概念"的附录，便于需要进行并行计算的读者对此有初步的了解.

(3) 学习本课程仍应加强上机计算实习，为此，新版增加了计算实习的题目，便于教学，教师可根据实际条件让学生选做其中的

3～5 题. 由于计算机算法语言发展很快,故不规定用哪种算法语言,目前我们向读者推荐的是集成化软件包 Matlab.

(4) 统一协调,改正错误. 本书第三版存在一些不协调之处和各种错误. 为保证新版质量,由李庆扬负责对全书整理加工,统一规格并改正旧版中的各种错误.

作者将新版"数值分析"交清华大学出版社重新出版,出版社委派曾多次使用本书的计算数学博士刘颖负责编辑加工,他不但改正了本书的一些错误并对本书修改提出了宝贵意见,提高了本书新版的质量,出版社还在较短时间使本书新版在开学前与读者见面,我们对清华大学出版社及刘颖博士表示衷心感谢.

作　者

2001 年 5 月

第三版说明

本书自1981年问世以来，为许多工科院校所采用，已先后出过两版，总发行量达四万余册. 1985年5月召开的工科院校计算数学教材评议会(南北会议)确认本书"基本符合应用数学专业的要求，可作为数值分析课程的教材，建议作者加以修改后重新出版". 我们遵照这次会议的建议和要求再次进行了修订. 新书在出版质量上有了显著的提高. 编者诚挚地感谢华中工学院出版社的同志们，为本书的重版付出了辛勤的劳动.

编 者

1986年12月

第二版前言

1980 年 7 月在大连召开的工科院校“应用数学专业教学学术会议”，根据教育部直属工科院校“应用数学专业教学计划”制定了“数值分析”课大纲，并决定由清华大学、华中工学院、浙江大学合编试用教材. 本书就是根据这次会议的决定编写的. 全书共分九章，第一、二、三章由李庆扬编写，第四、五、六章由王能超编写，第七、八、九章由易大义编写.

1981 年元月在杭州召开的工科院校计算数学第一次教材审稿会，对本教材初稿进行了审查，1982 年元月在上海交大召开的第二次计算数学教材审稿会，又对本书第一版提出了修改意见. 会议考虑到理工科院校各专业普遍开设“数值分析”课的情况，重新修订了大纲(72 学时). 本书第二版就是根据新大纲的要求修改的，它保持了第一版的主要内容及特点，但选材更注意基本要求，减少了部分内容，增加了部分习题答案. 本书可作为理工科院校应用数学、力学、物理、计算机软件等专业大学生及其他专业研究生“数值分析”(或“计算方法”)课的教材，也可供学习“计算方法”的科技工作者参考.

我们对参加两次审稿会的同志表示衷心感谢，他们以认真负责的态度对本书提出了许多宝贵意见，对提高教材质量起了很大作用.

编　者

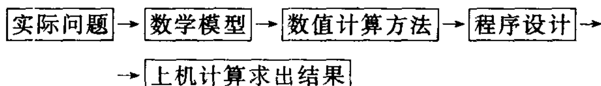
1982 年 7 月

目　录

第1章　绪　　论

1.1　数值分析研究对象与特点

数值分析是计算数学的一个主要部分，计算数学是数学科学的一个分支，它研究用计算机求解各种数学问题的数值计算方法及其理论与软件实现. 为了具体说明数值分析的研究对象，我们考察用计算机解决科学计算问题时经历的几个过程：



由实际问题的提出到上机求得问题解答的整个过程都可看作是应用数学的范畴. 如果细分的话，由实际问题应用有关科学知识和数学理论建立数学模型这一过程，通常作为应用数学的任务. 而根据数学模型提出求解的数值计算方法直到编出程序上机算出结果，这一过程则是计算数学的任务，也是数值分析研究的对象. 数值分析的内容包括函数的数值逼近、数值微分与数值积分、非线性方程数值解、数值线性代数、常微和偏微数值解等，它们都是以数学问题为研究对象的，只是它不像纯数学那样只研究数学本身的理论，而是把理论与计算紧密结合，着重研究数学问题的数值方法及其理论.

数值分析也称计算方法，但不应片面地理解为各种数值方法的简单罗列和堆积，同数学分析一样，它也是一门内容丰富，研究方法深刻，有自身理论体系的课程，既有纯数学高度抽象性与严密科学性的特点，又有应用的广泛性与实际试验的高度技术性的特

点，是一门与计算机使用密切结合的实用性很强的数学课程. 为了说明它与纯数学课的不同，例如考虑线性方程组数值解，在“线性代数”课程中只介绍解的存在唯一性及有关理论和精确解法，用这些理论和方法还不能在计算机上解上百个未知数的方程组，更不用说求解十几万个未知数的方程组了，要求解这类问题还应根据方程特点，研究适合计算机使用的，满足精度要求，计算时间省的有效算法及其相关的理论. 在实现这些算法时往往还要根据计算机的容量、字长、速度等指标，研究具体的求解步骤和程序设计技巧. 有的方法在理论上虽不够严格，但通过实际计算、对比分析等手段，证明是行之有效的方法，也应采用. 这些就是数值分析具有的特点，概括起来有四点：

第一，面向计算机，要根据计算机特点提供切实可行的有效算法. 即算法只能包括加、减、乘、除运算和逻辑运算，这些运算是计算机能直接处理的运算.

第二，有可靠的理论分析，能任意逼近并达到精度要求，对近似算法要保证收敛性和数值稳定性，还要对误差进行分析. 这些都建立在相应数学理论的基础上.

第三，要有好的计算复杂性，时间复杂性好是指节省时间，空间复杂性好是指节省存储量，这也是建立算法要研究的问题，它关系到算法能否在计算机上实现.

第四，要有数值实验，即任何一个算法除了从理论上要满足上述三点外，还要通过数值试验证明是行之有效的.

根据“数值分析”课程的特点，学习时我们首先要注意掌握方法的基本原理和思想，要注意方法处理的技巧及其与计算机的结合，要重视误差分析、收敛性及稳定性的基本理论；其次，要通过例子，学习使用各种数值方法解决实际计算问题；最后，为了掌握本课的内容，还应做一定数量的理论分析与计算练习. 由于本课内容包括了微积分、代数、常微分方程的数值方法，读者必须掌握这

几门课的基本内容才能学好这门课程.

1.2 数值计算的误差

1.2.1 误差来源与分类

用计算机解决科学计算问题首先要建立数学模型,它是对被描述的实际问题进行抽象、简化而得到的,因而是近似的. 我们把数学模型与实际问题之间出现的这种误差称为**模型误差**. 只有实际问题提法正确,建立数学模型时又抽象、简化得合理,才能得到好的结果. 由于这种误差难于用数量表示,通常都假定数学模型是合理的,这种误差可忽略不计,在"数值分析"中不予讨论. 在数学模型中往往还有一些根据观测得到的物理量,如温度、长度、电压等等,这些参量显然也包含误差. 这种由观测产生的误差称为**观测误差**,在"数值分析"中也不讨论这种误差. 数值分析只研究用数值方法求解数学模型产生的误差.

当数学模型不能得到精确解时,通常要用数值方法求它的近似解,其近似解与精确解之间的误差称为**截断误差**或**方法误差**. 例如,函数 $f(x)$用泰勒(Taylor)多项式

$$P_n(x) = f(0) + \frac{f'(0)}{1!}x + \frac{f''(0)}{2!}x^2 + \cdots + \frac{f^{(n)}(0)}{n!}x^n$$

近似代替,则数值方法的截断误差是

$$R_n(x) = f(x) - P_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}x^{n+1}, \xi \text{在} 0 \text{与} x \text{之间}.$$

有了求解数学问题的计算公式以后,用计算机做数值计算时,由于计算机的字长有限,原始数据在计算机上表示会产生误差,计算过程又可能产生新的误差,这种误差称为**舍入误差**. 例如,用 3.14159 近似代替 π,产生的误差

$$R = \pi - 3.14159 = 0.0000026\cdots$$

就是舍入误差.

此外由原始数据或机器中的十进制数转化为二进制数产生的初始误差对数值计算也将造成影响,分析初始数据的误差通常也归结为舍入误差.

研究计算结果的误差是否满足精度要求就是误差估计问题,本书主要讨论算法的截断误差与舍入误差,而截断误差将结合具体算法讨论. 为分析数值运算的舍入误差,先要对误差基本概念做简单介绍.

1.2.2 误差与有效数字

定义 1 设 x 为准确值,x^* 为 x 的一个近似值,称 $e^*=x^*-x$ 为近似值的**绝对误差**,简称**误差**.

注意这样定义的误差 e^* 可正可负,当绝对误差为正时近似值偏大,叫**强近似值**;当绝对误差为负时近似值偏小,叫**弱近似值**.

通常我们不能算出准确值 x,也不能算出误差 e^* 的准确值,只能根据测量工具或计算情况估计出误差的绝对值不超过某正数 ε^*,也就是误差绝对值的一个上界. ε^* 叫做近似值的**误差限**,它总是正数. 例如,用毫米刻度的米尺测量一长度 x,读出和该长度接近的刻度 x^*,x^* 是 x 的近似值,它的误差限是 0.5mm,于是 $|x^*-x|\leqslant 0.5\text{mm}$;如读出的长度为 765mm,则有 $|765-x|\leqslant 0.5$. 从这个不等式我们仍不知道准确的 x 是多少,但知道 $764.5\leqslant x\leqslant 765.5$,说明 x 在区间[764.5,765.5]内.

对于一般情形 $|x^*-x|\leqslant\varepsilon^*$,即

$$x^*-\varepsilon^*\leqslant x\leqslant x^*+\varepsilon^*,$$

这个不等式有时也表示为

$$x=x^*\pm\varepsilon^*.$$

误差限的大小还不能完全表示近似值的好坏. 例如,有两个量 $x=10\pm 1$,$y=1000\pm 5$,则

$$x^* = 10, \varepsilon_x^* = 1; y^* = 1000, \varepsilon_y^* = 5.$$

虽然 ε_y^* 比 ε_x^* 大 4 倍，但 $\varepsilon_y^*/y^* = \frac{5}{1000} = 0.5\%$ 比 $\varepsilon_x^*/x^* = \frac{1}{10} = 10\%$ 要小得多，这说明 y^* 近似 y 的程度比 x^* 近似 x 的程度要好得多. 所以，除考虑误差的大小外，还应考虑准确值 x 本身的大小. 我们把近似值的误差 e^* 与准确值 x 的比值

$$\frac{e^*}{x} = \frac{x^* - x}{x}$$

称为近似值 x^* 的**相对误差**，记作 e_r^*.

在实际计算中，由于真值 x 总是不知道的，通常取

$$e_r^* = \frac{e^*}{x^*} = \frac{x^* - x}{x^*}$$

作为 x^* 的相对误差，条件是 $e_r^* = \frac{e^*}{x^*}$ 较小，此时

$$\frac{e^*}{x} - \frac{e^*}{x^*} = \frac{e^*(x^* - x)}{x^* x} = \frac{(e^*)^2}{x^*(x^* - e^*)} = \frac{(e^*/x^*)^2}{1 - (e^*/x^*)}$$

是 e_r^* 的平方项级，故可忽略不计.

相对误差也可正可负，它的绝对值上界叫做**相对误差限**，记作 ε_r^*，即 $\varepsilon_r^* = \frac{\varepsilon^*}{|x^*|}$.

根据定义，上例中 $\frac{\varepsilon_x^*}{|x^*|} = 10\%$ 与 $\frac{\varepsilon_y^*}{|y^*|} = 0.5\%$ 分别为 x 与 y 的相对误差限，可见 y^* 近似 y 的程度比 x^* 近似 x 的程度好.

当准确值 x 有多位数时，常常按四舍五入的原则得到 x 的前几位近似值 x^*，例如

$$x = \pi = 3.14159265\cdots$$

取 3 位 $x_3^* = 3.14$，$\varepsilon_3^* \leqslant 0.002$，

取 5 位 $x_5^* = 3.1416$，$\varepsilon_5^* \leqslant 0.000008$，

它们的误差都不超过末位数字的半个单位，即

$$|\pi - 3.14| \leqslant \frac{1}{2} \times 10^{-2}, \qquad |\pi - 3.1416| \leqslant \frac{1}{2} \times 10^{-4}.$$

定义 2 若近似值 x^* 的误差限是某一位的半个单位，该位到 x^* 的第一位非零数字共有 n 位，就说 x^* 有 n 位**有效数字**. 它可表示为

$$x^* = \pm 10^m \times (a_1 + a_2 \times 10^{-1} + \cdots + a_n \times 10^{-(n-1)}), \tag{2.1}$$

其中 $a_i (i=1,\cdots,n)$ 是 0 到 9 中的一个数字，$a_1 \neq 0$，m 为整数，且

$$|x - x^*| \leqslant \frac{1}{2} \times 10^{m-n+1}. \tag{2.2}$$

如取 $x^* = 3.14$ 作 π 的近似值，x^* 就有 3 位有效数字，取 $x^* = 3.1416 \approx \pi$，x^* 就有 5 位有效数字.

例 1 按四舍五入原则写出下列各数具有 5 位有效数字的近似数：187.9325，0.03785551，8.000033，2.7182818.

按定义，上述各数具有 5 位有效数字的近似数分别是

187.93，0.037856，8.0000，2.7183.

注意 $x=8.000033$ 的 5 位有效数字近似数是 8.0000 而不是 8，因为 8 只有 1 位有效数字.

例 2 重力常数 g，如果以 $\mathrm{m/s^2}$ 为单位，$g \approx 9.80\mathrm{m/s^2}$；若以 $\mathrm{km/s^2}$ 为单位，$g \approx 0.00980\mathrm{km/s^2}$，它们都具有 3 位有效数字，因为按第一种写法

$$|g - 9.80| \leqslant \frac{1}{2} \times 10^{-2},$$

据(2.1)，这里 $m=0$，$n=3$；按第二种写法

$$|g - 0.00980| \leqslant \frac{1}{2} \times 10^{-5},$$

这里 $m=-3$，$n=3$. 它们虽然写法不同，但都具有 3 位有效数字. 至于绝对误差限，由于单位不同结果也不同，$\varepsilon_1^* = \frac{1}{2} \times 10^{-2} \mathrm{m/s^2}$，$\varepsilon_2^* = \frac{1}{2} \times 10^{-5} \mathrm{km/s^2}$，而相对误差都是

$$\varepsilon_r^* = 0.005/9.80 = 0.000005/0.00980.$$

注意相对误差与相对误差限是无量纲的，而绝对误差与误差限是有量纲的.

例 2 说明有效位数与小数点后有多少位数无关. 然而，从(2.2)可得到具有 n 位有效数字的近似数 x^*，其绝对误差限为

$$\varepsilon^* = \frac{1}{2} \times 10^{m-n+1},$$

在 m 相同的情况下，n 越大则 10^{m-n+1} 越小，故有效位数越多，绝对误差限越小.

至于有效数字与相对误差限的关系，有

定理 1 设近似数 x^* 表示为

$$x^* = \pm 10^m \times (a_1 + a_2 \times 10^{-1} + \cdots + a_l \times 10^{-(l-1)}), \tag{2.1$'$}$$

其中 $a_i(i=1,2,\cdots,l)$ 是 0 到 9 中的一个数字，$a_1 \neq 0$，m 为整数. 若 x^* 具有 n 位有效数字，则其相对误差限为

$$\varepsilon_r^* \leqslant \frac{1}{2a_1} \times 10^{-(n-1)};$$

反之，若 x^* 的相对误差限 $\varepsilon_r^* \leqslant \frac{1}{2(a_1+1)} \times 10^{-(n-1)}$，则 x^* 至少具有 n 位有效数字.

证明 由(2.1)$'$可得

$$a_1 \times 10^m \leqslant |x^*| < (a_1 + 1) \times 10^m,$$

当 x^* 有 n 位有效数字时

$$\varepsilon_r^* = \frac{|x - x^*|}{|x^*|} \leqslant \frac{0.5 \times 10^{m-n+1}}{a_1 \times 10^m} = \frac{1}{2a_1} \times 10^{-n+1};$$

反之，由

$$|x - x^*|$$

$$= |x^*| \varepsilon_r^* < (a_1 + 1) \times 10^m \times \frac{1}{2(a_1 + 1)} \times 10^{-n+1}$$

$$= 0.5 \times 10^{m-n+1},$$

故 x^* 至少有 n 位有效数字．定理证完．

定理说明，有效位数越多，相对误差限越小．

例 3　要使$\sqrt{20}$的近似值的相对误差限小于 0.1%，要取几位有效数字？

设取 n 位有效数字，由定理 1，$\varepsilon_r^* \leqslant \frac{1}{2a_1} \times 10^{-n+1}$．由于$\sqrt{20} = 4.4\cdots$，知 $a_1 = 4$，故只要取 $n=4$，就有

$$\varepsilon_r^* \leqslant 0.125 \times 10^{-3} < 10^{-3} = 0.1\%,$$

即只要对$\sqrt{20}$的近似值取 4 位有效数字，其相对误差限就小于 0.1%．此时由开方表得$\sqrt{20} \approx 4.472$．

1.2.3　数值运算的误差估计

两个近似数 x_1^* 与 x_2^*，其误差限分别为 $\varepsilon(x_1^*)$ 及 $\varepsilon(x_2^*)$，它们进行加、减、乘、除运算得到的误差限分别为

$$\varepsilon(x_1^* \pm x_2^*) = \varepsilon(x_1^*) + \varepsilon(x_2^*);$$

$$\varepsilon(x_1^* x_2^*) \approx |x_1^*| \varepsilon(x_2^*) + |x_2^*| \varepsilon(x_1^*);$$

$$\varepsilon(x_1^* / x_2^*) \approx \frac{|x_1^*| \varepsilon(x_2^*) + |x_2^*| \varepsilon(x_1^*)}{|x_2^*|^2} \qquad (x_2^* \neq 0).$$

更一般的情况是，当自变量有误差时计算函数值也产生误差，其误差限可利用函数的泰勒展开式进行估计．设 $f(x)$ 是一元函数，x 的近似值为 x^*，以 $f(x^*)$ 近似 $f(x)$，其误差界记作 $\varepsilon(f(x^*))$，可用泰勒展开

$$f(x) - f(x^*) = f'(x^*)(x - x^*) + \frac{f''(\xi)}{2}(x - x^*)^2,$$

ξ 介于 x, x^* 之间，

取绝对值得

$$|f(x) - f(x^*)| \leqslant |f'(x^*)| \varepsilon(x^*) + \frac{|f''(\xi)|}{2} \varepsilon^2(x^*).$$

假定 $f'(x^*)$ 与 $f''(x^*)$ 的比值不太大，可忽略 $\varepsilon(x^*)$ 的高阶项，于是可得计算函数的误差限

$$\varepsilon(f(x^*)) \approx |f'(x^*)| \varepsilon(x^*).$$

当 f 为多元函数时，例如计算 $A=f(x_1,\cdots,x_n)$. 如果 $x_1,\cdots,x_n$ 的近似值为 $x_1^*,\cdots,x_n^*$，则 A 的近似值为 $A^*=f(x_1^*,\cdots,x_n^*)$，于是由泰勒展开得函数值 A^* 的误差 $e(A^*)$ 为

$$\begin{aligned} e(A^*) = A^* - A &= f(x_1^*,\cdots,x_n^*) - f(x_1,\cdots,x_n) \\ &\approx \sum_{k=1}^{n} \left(\frac{\partial f(x_1^*,\cdots,x_n^*)}{\partial x_k} \right) (x_k^* - x_k) \\ &= \sum_{k=1}^{n} \left(\frac{\partial f}{\partial x_k} \right)^* e_k^*, \end{aligned}$$

于是误差限

$$\varepsilon(A^*) \approx \sum_{k=1}^{n} \left| \left(\frac{\partial f}{\partial x_k} \right)^* \right| \varepsilon(x_k^*); \tag{2.3}$$

而 A^* 的相对误差限为

$$\varepsilon_r^* = \varepsilon_r(A^*) = \frac{\varepsilon(A^*)}{|A^*|} \approx \sum_{k=1}^{n} \left| \left(\frac{\partial f}{\partial x_k} \right)^* \right| \frac{\varepsilon(x_k^*)}{|A^*|}. \tag{2.4}$$

例 4 已测得某场地长 l 的值为 $l^*=110\text{m}$，宽 d 的值为 $d^*=80\text{m}$，已知 $|l-l^*|\leqslant 0.2\text{m}$，$|d-d^*|\leqslant 0.1\text{m}$. 试求面积 $s=ld$ 的绝对误差限与相对误差限.

解 因 $s=ld$，$\frac{\partial s}{\partial l}=d$，$\frac{\partial s}{\partial d}=l$，由(2.3)知

$$\varepsilon(s^*) \approx \left| \left(\frac{\partial s}{\partial l} \right)^* \right| \varepsilon(l^*) + \left| \left(\frac{\partial s}{\partial d} \right)^* \right| \varepsilon(d^*),$$

其中

$$\left(\frac{\partial s}{\partial l} \right)^* = d^* = 80\text{m}, \qquad \left(\frac{\partial s}{\partial d} \right)^* = l^* = 110\text{m},$$

而 $\qquad \varepsilon(l^*)=0.2\text{m}, \qquad \varepsilon(d^*)=0.1\text{m},$

于是绝对误差限

$$\varepsilon(s^*)\approx 80\times(0.2)+110\times(0.1)=27(\mathrm{m}^2);$$

相对误差限

$$\varepsilon_r(s^*)=\frac{\varepsilon(s^*)}{|s^*|}=\frac{\varepsilon(s^*)}{l^*d^*}\approx\frac{27}{8800}=0.31\%.$$

1.3 误差定性分析与避免误差危害

数值运算中的误差分析是个很重要而复杂的问题，上节讨论了不精确数据运算结果的误差限，它只适用于简单情形，然而一个工程或科学计算问题往往要运算千万次，由于每步运算都有误差，如果每步都做误差分析是不可能的，也不科学，因为误差积累有正有负，绝对值有大有小，都按最坏情况估计误差限得到的结果比实际误差大得多，这种保守的误差估计不反映实际误差积累. 考虑到误差分布的随机性，有人用概率统计方法，将数据和运算中的舍入误差视为适合某种分布的随机变量，然后确定计算结果的误差分布，这样得到的误差估计更接近实际，这种方法称为**概率分析法**.

20 世纪 60 年代以后对舍入误差分析提出了一些新方法，较重要的有以下两种：

1. 向后误差分析法是把新算出的量由某个公式表达，它仅含基本算术运算，如假定 $a_1,\cdots,a_n$ 是前面已算出的量或原始数据，新算出量

$$x=g(a_1,\cdots,a_n).$$

若 a_i 的摄动为 ε_i，使得由浮点运算得出结果为

$$x_{fl}=g(a_1+\varepsilon_1,\cdots,a_n+\varepsilon_n).$$

则可根据 ε_i 的界由摄动理论估计最后舍入误差 $|x-x_{fl}|$ 的界，威克逊(Wilkinson)将这种方法应用于数值代数(矩阵运算)的误差分析，取得较好效果.

2. 区间分析法是把参加运算的数 $x,y,z,\cdots$ 都看成区间量 $X,Y,Z,\cdots$，根据区间运算规则求得最后结果的近似值及误差限. 例如，x,y 的近似数为 α,β，由于 $|x-\alpha|\leqslant\delta\alpha$，$|y-\beta|\leqslant\delta\beta$，则 $x\in[\alpha-\delta\alpha,\alpha+\delta\alpha]=X$，$y\in[\beta-\delta\beta,\beta+\delta\beta]=Y$，若计算 $z=x*y$（$*$ 为运算符号），由 $\underline{Z}=\underline{X}*\underline{Y}=[\underline{z},\bar{z}]=[z-\delta z,z+\delta z]$，则 z 为所求近似值，而 δz 则为误差限.

上面简略介绍了误差分析的几种方法，但都不是十分有效的，目前尚无有效的方法对误差做出定量估计. 为了确保数值计算结果的正确性，首先需对数值计算问题做定性分析，为此本节讨论以下三个问题.

1.3.1 病态问题与条件数

对一个数值问题本身如果输入数据有微小扰动（即误差），引起输出数据（即问题解）相对误差很大，这就是病态问题. 例如计算函数值 $f(x)$ 时，若 x 有扰动 $\Delta x=x-x^*$，其相对误差为 $\frac{\Delta x}{x}$，函数值 $f(x^*)$ 的相对误差为 $\frac{f(x)-f(x^*)}{f(x)}$. 相对误差比值

$$\left|\frac{f(x)-f(x^*)}{f(x)}\right|\Big/\left|\frac{\Delta x}{x}\right|\approx\left|\frac{xf'(x)}{f(x)}\right|=C_p, \qquad (3.1)$$

C_p 称为计算函数值问题的条件数. 自变量相对误差一般不会太大，如果条件数 C_p 很大，将引起函数值相对误差很大，出现这种情况的问题就是病态问题.

例如，$f(x)=x^n$，则有 $C_p=n$，它表示相对误差可能放大 n 倍. 如 $n=10$，有 $f(1)=1$，$f(1.02)\approx1.24$，若取 $x=1$，$x^*=1.02$ 自变量相对误差为 2%，函数值相对误差为 24%，这时问题可以认为是病态的. 一般情况条件数 $C_p\geqslant10$ 就认为是病态，C_p 越大病态越严重.

其他计算问题也要分析是否病态. 例如解线性方程组,如果输入数据有微小误差引起解的巨大误差,就认为是病态方程组,我们将在第5章用矩阵的条件数来分析这种现象.

1.3.2 算法的数值稳定性

用一个算法进行计算,由于初始数据误差在计算中传播使计算结果误差增长很快就是数值不稳定的,先看下例.

例 5 计算 $I_n = e^{-1}\int_0^1 x^n e^x \mathrm{d}x (n = 0,1,\cdots)$ 并估计误差.

由分部积分可得计算 I_n 的递推公式

$$I_n = 1 - nI_{n-1} \qquad (n = 1,2,\cdots), \tag{3.2}$$

$$I_0 = e^{-1}\int_0^1 e^x \mathrm{d}x = 1 - e^{-1}.$$

若计算出 I_0,代入(3.2),可逐次求出 $I_1, I_2, \cdots$ 的值. 要算出 I_0 就要先计算 e^{-1},若用泰勒多项式展开部分和

$$e^{-1} \approx 1 + (-1) + \frac{(-1)^2}{2!} + \cdots + \frac{(-1)^k}{k!},$$

并取 $k=7$,用4位小数计算,则得 $e^{-1}\approx 0.3679$,截断误差 $R_7 = |e^{-1} - 0.3679| \leqslant \frac{1}{8!} < \frac{1}{4}\times 10^{-4}$. 计算过程中小数点后第5位的数字按四舍五入原则舍入,由此产生的舍入误差这里先不讨论. 当初值取为 $I_0 \approx 0.6321 = \tilde{I}_0$ 时,用(3.2)递推的计算公式为

$$(\mathrm{A})\begin{cases} \tilde{I}_0 = 0.6321; \\ \tilde{I}_n = 1 - n\tilde{I}_{n-1} \qquad (n = 1,2,\cdots). \end{cases}$$

计算结果见表1-1的 $\tilde{I}_n$ 列. 用 $\tilde{I}_0$ 近似 I_0 产生的误差 $E_0 = I_0 - \tilde{I}_0$ 就是初值误差,它对后面计算结果是有影响的.

表 1-1

n	$\tilde{I}_n$(用(A)算)	I^*(用(B)算)	n	$\tilde{I}_n$(用(A)算)	I^*(用(B)算)
0	0.6321 ↓	0.6321	5	0.1480 ↓	0.1455
1	0.3679	0.3679	6	0.1120	0.1268
2	0.2642	0.2643	7	0.2160	0.1121
3	0.2074	0.2073 ↑	8	−0.7280	0.1035 ↑
4	0.1704	0.1708	9	7.552	0.0684

从表中看到 $\tilde{I}_8$ 出现负值，这与一切 $I_n>0$ 相矛盾. 实际上，由积分估值得

$$\frac{e^{-1}}{n+1}=e^{-1}\left(\min_{0\leqslant x\leqslant 1} e^x\right)\int_0^1 x^n\mathrm{d}x<I_n<e^{-1}\left(\max_{0\leqslant x\leqslant 1} e^x\right)\int_0^1 x^n\mathrm{d}x$$

$$=\frac{1}{n+1}. \tag{3.3}$$

因此，当 n 较大时，用 $\tilde{I}_n$ 近似 I_n 显然是不正确的. 这里计算公式与每步计算都是正确的，那么是什么原因使计算结果错误呢？主要就是初值 $\tilde{I}_0$ 有误差 $E_0=I_0-\tilde{I}_0$，由此引起以后各步计算的误差 $E_n=I_n-\tilde{I}_n$ 满足关系

$$E_n=-nE_{n-1}\qquad (n=1,2,\cdots).$$

容易推得

$$E_n=(-1)^n n!E_0,$$

这说明 $\tilde{I}_0$ 有误差 E_0，则 $\tilde{I}_n$ 就是 E_0 的 $n!$ 倍误差. 例如，$n=8$，若 $|E_0|=\frac{1}{2}\times 10^{-4}$，则 $|E_8|=8!\times|E_0|>2$. 这就说明 $\tilde{I}_8$ 完全不能近似 I_8 了. 它表明计算公式(A)是数值不稳定的.

我们现在换一种计算方案. 由(3.3)取 $n=9$，取

$$\frac{e^{-1}}{10}<I_9<\frac{1}{10},$$

我们粗略取 $I_9\approx\frac{1}{2}\left(\frac{1}{10}+\frac{e^{-1}}{10}\right)=0.0684=I_9$，然后将公式(3.2)倒

过来算,即由 I_9^* 算出 $I_8^*, I_7^*, \cdots, I_9^*$,公式为

$$
(\mathrm{B})\begin{cases} I_9^* = 0.0684, \\ I_{n-1}^* = \dfrac{1}{n}(1 - I_n^*) \qquad\qquad (n = 9, 8, \cdots, 1); \end{cases}
$$

计算结果见表 1-1 的 I_n^* 列. 我们发现 I_0^* 与 I_0 的误差不超过 10^{-4}. 记 $E_n^* = I_n - I_n^*$,则 $|E_0^*| = \dfrac{1}{n!}|E_n^*|$,$E_0^*$ 比 E_n^* 缩小了 $n!$ 倍,因此,尽管 E_9^* 较大,但由于误差逐步缩小,故可用 I_n^* 近似 I_n. 反之,当用方案(A)计算时,尽管初值 $\tilde{I}_0$ 相当准确,由于误差传播是逐步扩大的,因而计算结果不可靠. 此例说明,数值不稳定的算法是不能使用的.

定义 3 一个算法如果输入数据有误差,而在计算过程中舍入误差不增长,则称此算法是数值稳定的,否则称此算法为不稳定的.

在例 5 中算法(B)是数值稳定的,而算法(A)是不稳定的. 数值不稳定现象属于误差危害现象,如何防止误差危害下面将进一步讨论.

1.3.3 避免误差危害的若干原则

数值计算中首先要分清问题是否病态和算法是否数值稳定,计算时还应尽量避免误差危害,防止有效数字的损失,下面给出若干原则.

1. 要避免除数绝对值远远小于被除数绝对值的除法

用绝对值小的数作除数舍入误差会增大,如计算 $\dfrac{x}{y}$,若 $0 < |y| \ll |x|$,则可能对计算结果带来严重影响,应尽量避免.

例 6 线性方程组

$$
\begin{cases} 0.00001x_1 + x_2 = 1, \\ 2x_1 + x_2 = 2. \end{cases}
$$

的准确解为

$$x_1 = \frac{200000}{399999} = 0.50000125, \qquad x_2 = \frac{199998}{199999} = 0.999995.$$

现在四位浮点十进制数(仿机器实际计算)下用消去法求解,上述方程写成

$$\begin{cases} 10^{-4} \cdot 0.1000x_1 + 10^1 \cdot 0.1000x_2 = 10^1 \cdot 0.1000, \\ 10^1 \cdot 0.2000x_1 + 10^1 \cdot 0.1000x_2 = 10^1 \cdot 0.2000. \end{cases}$$

若用$\frac{1}{2}(10^{-4} \cdot 0.1000)$除第一方程减第二方程,则出现用小的数除大的数,得到

$$\begin{cases} 10^{-4} \cdot 0.1000x_1 + 10^1 \cdot 0.1000x_2 = 10^1 \cdot 0.1000, \\ 10^5 \cdot 0.2000x_2 = 10^6 \cdot 0.2000. \end{cases}$$

由此解出

$$x_1 = 0, \qquad x_2 = 10^1 \cdot 0.1000 = 1,$$

显然严重失真.

若反过来用第二个方程消去第一个方程中含 x_1 的项,则避免了大数被小数除,得到

$$\begin{cases} 10^6 \cdot 0.1000x_2 = 10^6 \cdot 0.1000, \\ 10^1 \cdot 0.2000x_1 + 10^1 \cdot 0.1000x_2 = 10^1 \cdot 0.2000. \end{cases}$$

由此求得相当好的近似解

$$x_1 = 0.5000, \qquad x_2 = 10^1 \cdot 0.1000.$$

2. 要避免两相近数相减

在数值计算中两相近数相减有效数字会严重损失. 例如,$x=532.65$,$y=532.52$ 都具有五位有效数字,但 $x-y=0.13$ 只有两位有效数字. 这说明必须尽量避免出现这类运算. 最好是改变计算方法,防止这种现象产生. 现举例说明.

例 7 求 $x^2-16x+1=0$ 的小正根.

解 $x_1=8+\sqrt{63}$,$x_2=8-\sqrt{63}\approx 8-7.94=0.06=x_2^*$,

x_2^* 只有一位有效数字. 若改用

$$x_2 = 8-\sqrt{63} = \frac{1}{8+\sqrt{63}} \approx \frac{1}{15.94} \approx 0.0627$$

具有 3 位有效数字.

例 8 计算 $A=10^7(1-\cos 2°)$(用四位数学用表).

由于 $\cos 2°=0.9994$,直接计算

$$A = 10^7(1-\cos 2°) = 10^7(1-0.9994) = 6\times 10^3.$$

只有一位有效数字. 若利用 $1-\cos x=2\sin^2\frac{x}{2}$,则

$$A = 10^7(1-\cos 2°) = 2\times(\sin 1°)^2\times 10^7 = 6.13\times 10^3$$

具有三位有效数字(这里 $\sin 1°=0.0175$).

此例说明,可通过改变计算公式避免或减少有效数字的损失. 类似地,如果 x_1 和 x_2 很接近时,则

$$\lg x_1 - \lg x_2 = \lg\frac{x_1}{x_2}.$$

用右边算式有效数字就不损失. 当 x 很大时,

$$\sqrt{x+1}-\sqrt{x} = \frac{1}{\sqrt{x+1}+\sqrt{x}},$$

都用右端算式代替左端. 一般情况,当 $f(x)\approx f(x^*)$时,可用泰勒展开

$$f(x)-f(x^*) = f'(x^*)(x-x^*)+\frac{f''(x^*)}{2}(x-x^*)^2+\cdots$$

取右端的有限项近似左端. 如果无法改变算式,则采用增加有效位数进行运算;在计算机上则采用双倍字长运算,但这要增加机器计算时间和多占内存单元.

3. 要防止大数"吃掉"小数

在数值运算中参加运算的数有时数量级相差很大,而计算机位数有限,如不注意运算次序就可能出现大数"吃掉"小数的现象,影响计算结果的可靠性.

例 9 在五位十进制计算机上，计算

$$A = 52492 + \sum_{i=1}^{1000} \delta_i,$$

其中 $0.1 \leqslant \delta_i \leqslant 0.9$.

把运算的数写成规格化形式

$$A = 0.52492 \times 10^5 + \sum_{i=1}^{1000} \delta_i.$$

由于在计算机内计算时要对阶，若取 $\delta_i = 0.9$，对阶时 $\delta_i = 0.000009 \times 10^5$，在五位的计算机中表示为机器 0，因此

$$A = 0.52492 \times 10^5 + 0.000009 \times 10^5 + \cdots + 0.000009 \times 10^5$$

$\triangleq 0.52492 \times 10^5$（符号 $\triangleq$ 表示机器中相等），

结果显然不可靠，这是由于运算中出现了大数 52492"吃掉"小数 δ_i 造成的. 如果计算时先把数量级相同的一千个 δ_i 相加，最后再加 52492，就不会出现大数"吃"小数现象，这时

$$0.1 \times 10^3 \leqslant \sum_{i=1}^{1000} \delta_i \leqslant 0.9 \times 10^3,$$

于是

$$0.001 \times 10^5 + 0.52492 \times 10^5 \leqslant A \leqslant 0.009 \times 10^5 + 0.52492 \times 10^5,$$

$$52592 \leqslant A \leqslant 53392.$$

4. 注意简化计算步骤，减少运算次数

同样一个计算问题，如果能减少运算次数，不但可节省计算机的计算时间，还能减少舍入误差. 这是数值计算必须遵从的原则，也是"数值分析"要研究的重要内容.

例 10 计算多项式

$$P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

的值，若直接计算 $a_k x^k$ 再逐项相加，一共需做

$$n + (n-1) + \cdots + 2 + 1 = \frac{n(n+1)}{2}$$

次乘法和 n 次加法. 若采用秦九韶算法

$$\begin{cases} S_n = a_n, \\ S_k = xS_{k+1} + a_k (k = n-1, \cdots, 2, 1, 0), \\ P_n(x) = S_0. \end{cases} \tag{3.4}$$

只要 n 次乘法和 n 次加法就可算出 $P_n(x)$ 的值.

在"数值分析"中,这种节省计算次数的算法还有不少. 本书第 3 章介绍的 FFT 算法,就是一个最成功的范例.

评 注

本章第 1 节简要地介绍了数值分析的研究对象,它是计算数学的主要部分,关于计算数学介绍可参考《中国大百科全书 · 数学》中的有关条目. 第 2 和第 3 节中介绍了误差的基本概念与误差分析的若干原则,数值计算中的舍入误差是一个困难而复杂的问题,目前尚无真正有效的定量估计方法,第 3 节中提到的向后误差分析法(见文献[8])及区间分析法[9],实际计算时仍有不少困难,对大型科学计算的误差估计仍难于使用. 因此在本书中更着重对误差的定性分析,即对每个具体算法只要是数值稳定的,就不必再做舍入误差估计,至于方法的截断误差将结合不同问题的具体算法进行讨论.

习 题

1. 设 $x>0$,x 的相对误差为 δ,求 $\ln x$ 的误差.

2. 设 x 的相对误差为 2%,求 x^n 的相对误差.

3. 下列各数都是经过四舍五入得到的近似数,即误差限不超过最后一位的半个单位,试指出它们是几位有效数字:

$$x_1^* = 1.1021,\quad x_2^* = 0.031,\quad x_3^* = 385.6,$$
$$x_4^* = 56.430,\quad x_5^* = 7\times 1.0.$$

4. 利用公式(2.3)求下列各近似值的误差限：

(i) $x_1^* + x_2^* + x_4^*$，　(ii) $x_1^* x_2^* x_3^*$，　(iii) x_2^* / x_4^*.

其中 $x_1^*, x_2^*, x_3^*, x_4^*$ 均为第 3 题所给的数.

5. 计算球体积要使相对误差限为 1%，问度量半径 R 时允许的相对误差限是多少？

6. 设 $Y_0 = 28$，按递推公式

$$Y_n = Y_{n-1} - \frac{1}{100}\sqrt{783}\qquad (n = 1,2,\cdots)$$

计算到 Y_{100}. 若取 $\sqrt{783} \approx 27.982$(5 位有效数字)，试问计算 Y_{100} 将有多大误差？

7. 求方程 $x^2 - 56x + 1 = 0$ 的两个根，使它至少具有 4 位有效数字($\sqrt{783} \approx 27.982$).

8. 当 N 充分大时，怎样求 $\int_N^{N+1} \frac{1}{1+x^2}\mathrm{d}x$？

9. 正方形的边长大约为 100cm，应怎样测量才能使其面积误差不超过 $1\mathrm{cm}^2$？

10. 设 $S = \frac{1}{2}gt^2$，假定 g 是准确的，而对 t 的测量有 ± 0.1 秒的误差，证明当 t 增加时 S 的绝对误差增加，而相对误差却减少.

11. 序列 $\{y_n\}$ 满足递推关系

$$y_n = 10y_{n-1} - 1\qquad (n = 1,2,\cdots),$$

若 $y_0 = \sqrt{2} \approx 1.41$(三位有效数字)，计算到 y_{10} 时误差有多大？这个计算过程稳定吗？

12. 计算 $f = (\sqrt{2} - 1)^6$，取 $\sqrt{2} \approx 1.4$，利用下列等式计算，哪一个得到的结果最好？

$$\frac{1}{(\sqrt{2}+1)^6},\qquad (3-2\sqrt{2})^3,$$
$$\frac{1}{(3+2\sqrt{2})^3},\qquad 99 - 70\sqrt{2}.$$

13. $f(x)=\ln(x-\sqrt{x^2-1})$，求 $f(30)$ 的值. 若开平方用6位函数表，问求对数时误差有多大？若改用另一等价公式

$$\ln(x-\sqrt{x^2-1})=-\ln(x+\sqrt{x^2-1})$$

计算，求对数时误差有多大？

第2章 插 值 法

2.1 引 言

许多实际问题都用函数 $y=f(x)$ 来表示某种内在规律的数量关系，其中相当一部分函数是通过实验或观测得到的. 虽然 $f(x)$ 在某个区间 $[a,b]$ 上是存在的，有的还是连续的，但却只能给出 $[a,b]$ 上一系列点 x_i 的函数值 $y_i=f(x_i)$ $(i=0,1,\cdots,n)$，这只是一张函数表. 有的函数虽有解析表达式，但由于计算复杂，使用不方便，通常也造一个函数表，如大家熟悉的三角函数表、对数表、平方根和立方根表等等. 为了研究函数的变化规律，往往需要求出不在表上的函数值. 因此，我们希望根据给定的函数表做一个既能反映函数 $f(x)$ 的特性，又便于计算的简单函数 $P(x)$，用 $P(x)$ 近似 $f(x)$. 通常选一类较简单的函数（如代数多项式或分段代数多项式）作为 $P(x)$，并使 $P(x_i)=f(x_i)$ 对 $i=0,1,\cdots,n$ 成立. 这样确定的 $P(x)$ 就是我们希望得到的插值函数. 例如，在现代机械工业中用计算机程序控制加工机械零件，根据设计可给出零件外形曲线的某些型值点 (x_i,y_i) $(i=0,1,\cdots,n)$，加工时为控制每步走刀方向及步数，就要算出零件外形曲线其他点的函数值，才能加工出外表光滑的零件，这就是求插值函数的问题. 下面我们给出有关插值法的定义.

设函数 $y=f(x)$ 在区间 $[a,b]$ 上有定义，且已知在点 $a\leqslant x_0<x_1<\cdots<x_n\leqslant b$ 上的值 $y_0,y_1,\cdots,y_n$，若存在一简单函数 $P(x)$，使

$$P(x_i)=y_i \qquad (i=0,1,\cdots,n) \tag{1.1}$$

成立，就称 $P(x)$ 为 $f(x)$ 的**插值函数**，点 $x_0,x_1,\cdots,x_n$ 称为**插值节**

点,包含插值节点的区间$[a,b]$称为**插值区间**,求插值函数 $P(x)$的方法称为**插值法**. 若 $P(x)$是次数不超过 n 的代数多项式,即

$$P(x)=a_0+a_1x+\cdots+a_nx^n, \tag{1.2}$$

其中 a_i 为实数,就称 $P(x)$为**插值多项式**,相应的插值法称为**多项式插值**. 若 $P(x)$为分段的多项式,就称为**分段插值**. 若 $P(x)$为三角多项式,就称为**三角插值**. 本章只讨论多项式插值与分段插值.

从几何上看,插值法就是求曲线 $y=P(x)$,使其通过给定的 $n+1$个点(x_i,y_i),$i=0,1,\cdots,n$,并用它近似已知曲线 $y=f(x)$,见图 2-1.

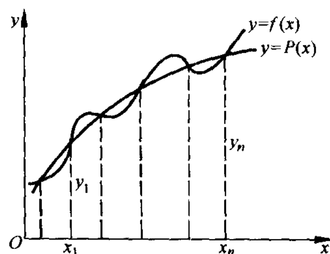


图 2-1

插值法是一种古老的数学方法,它来自生产实践. 早在一千多年前,我国科学家在研究历法上就应用了线性插值与二次插值,但它的基本理论和结果却是在微积分产生以后才逐步完善的,其应用也日益增多,特别是在电子计算机广泛使用以后,由于航空、造船、精密机械加工等实际问题的需要,使插值法在实践上或理论上显得更为重要,并得到进一步发展,尤其是近几十年发展起来的样条(spline)插值,更获得了广泛的应用.

本章主要研究如何求出插值多项式,分段插值函数,样条插值函数;讨论插值多项式 $P(x)$的存在唯一性、收敛性及误差估计等.

2.2 拉格朗日插值

2.2.1 线性插值与抛物插值

对给定的插值点为求得形如(1.2)的插值多项式可以有各种不同方法，下面先讨论 $n=1$ 的简单情形，假定给定区间 $[x_k, x_{k+1}]$ 及端点函数值 $y_k=f(x_k)$，$y_{k+1}=f(x_{k+1})$，要求线性插值多项式 $L_1(x)$，使它满足

$$L_1(x_k)=y_k,\ L_1(x_{k+1})=y_{k+1}.$$

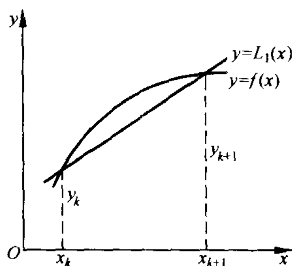


图 2-2

$y=L_1(x)$ 的几何意义就是通过两点 (x_k, y_k) 与 (x_{k+1}, y_{k+1}) 的直线，如图 2-2 所示，$L_1(x)$ 的表达式可由几何意义直接给出

$$L_1(x)=y_k+\frac{y_{k+1}-y_k}{x_{k+1}-x_k}(x-x_k)\quad(\text{点斜式}),$$

$$L_1(x)=\frac{x_{k+1}-x}{x_{k+1}-x_k}y_k+\frac{x-x_k}{x_{k+1}-x_k}y_{k+1}(\text{两点式}).\tag{2.1}$$

由两点式看出，$L_1(x)$ 是由两个线性函数

$$l_k(x)=\frac{x-x_{k+1}}{x_k-x_{k+1}},\qquad l_{k+1}(x)=\frac{x-x_k}{x_{k+1}-x_k}\tag{2.2}$$

的线性组合得到，其系数分别为 y_k 及 y_{k+1}，即

$$L_1(x) = y_k l_k(x) + y_{k+1} l_{k+1}(x). \tag{2.3}$$

显然，$l_k(x)$及 $l_{k+1}(x)$也是线性插值多项式，在节点 x_k 及 x_{k+1} 上满足条件

$$l_k(x_k) = 1, \qquad l_k(x_{k+1}) = 0;$$
$$l_{k+1}(x_k) = 0, \qquad l_{k+1}(x_{k+1}) = 1.$$

我们称函数 $l_k(x)$及 $l_{k+1}(x)$为**线性插值基函数**，它们的图形见图 2-3.

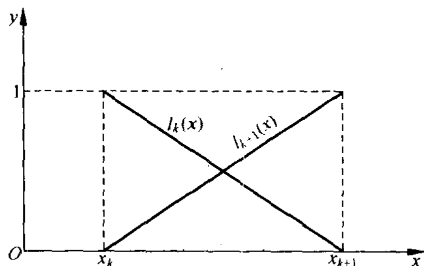


图　2-3

下面讨论 $n=2$ 的情况. 假定插值节点为 x_{k-1}, x_k, x_{k+1}，要求二次插值多项式 $L_2(x)$，使它满足

$$L_2(x_j) = y_j \qquad (j = k-1, k, k+1).$$

我们知道 $y=L_2(x)$在几何上就是通过三点(x_{k-1}, y_{k-1})，(x_k, y_k)，(x_{k+1}, y_{k+1})的抛物线. 为了求出 $L_2(x)$的表达式，可采用基函数方法，此时基函数 $l_{k-1}(x)$，$l_k(x)$及 $l_{k+1}(x)$是二次函数，且在节点上满足条件

$$\left.\begin{aligned} &l_{k-1}(x_{k-1}) = 1, \quad l_{k-1}(x_j) = 0 \quad (j = k, k+1); \\ &l_k(x_k) = 1, \quad l_k(x_j) = 0 \quad (j = k-1, k+1); \\ &l_{k+1}(x_{k+1}) = 1, \quad l_{k+1}(x_j) = 0 \quad (j = k-1, k). \end{aligned}\right\} \tag{2.4}$$

满足条件(2.4)的插值基函数是很容易求出的，例如求 $l_{k-1}(x)$，因它有两个零点 x_k 及 x_{k+1}，故可表示为

$$l_{k-1}(x) = A(x-x_k)(x-x_{k+1}),$$

其中 A 为待定系数,可由条件 $l_{k-1}(x_{k-1})=1$ 定出

$$A = \frac{1}{(x_{k-1}-x_k)(x_{k-1}-x_{k+1})},$$

于是 $$l_{k-1}(x) = \frac{(x-x_k)(x-x_{k+1})}{(x_{k-1}-x_k)(x_{k-1}-x_{k+1})}.$$

同理可得 $$l_k(x) = \frac{(x-x_{k-1})(x-x_{k+1})}{(x_k-x_{k-1})(x_k-x_{k+1})},$$

$$l_{k+1}(x) = \frac{(x-x_{k-1})(x-x_k)}{(x_{k+1}-x_{k-1})(x_{k+1}-x_k)}.$$

二次插值基函数 $l_{k-1}(x), l_k(x), l_{k+1}(x)$ 在区间 $[x_{k-1}, x_{k+1}]$ 上的图形见图 2-4.

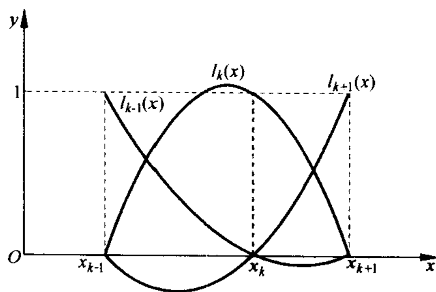


图 2-4

利用二次插值基函数 $l_{k-1}(x), l_k(x), l_{k+1}(x)$,立即得到二次插值多项式

$$L_2(x) = y_{k-1}l_{k-1}(x) + y_k l_k(x) + y_{k+1}l_{k+1}(x), \qquad (2.5)$$

显然,它满足条件 $L_2(x_j)=y_j\ (j=k-1,k,k+1)$. 将上面求得的 $l_{k-1}(x), l_k(x), l_{k+1}(x)$ 代入(2.5),得

$$L_2(x) = y_{k-1}\frac{(x-x_k)(x-x_{k+1})}{(x_{k-1}-x_k)(x_{k-1}-x_{k+1})}$$

$$+ y_k \frac{(x - x_{k-1})(x - x_{k+1})}{(x_k - x_{k-1})(x_k - x_{k+1})}$$

$$+ y_{k+1} \frac{(x - x_{k-1})(x - x_k)}{(x_{k+1} - x_{k-1})(x_{k+1} - x_k)}.$$

2.2.2 拉格朗日插值多项式

上面我们对 $n=1$ 及 $n=2$ 的情况，得到了一次与二次插值多项式 $L_1(x)$ 及 $L_2(x)$，它们分别由(2.3)与(2.5)表示. 这种用插值基函数表示的方法容易推广到一般情形. 下面讨论如何构造通过 $n+1$ 个节点 $x_0<x_1<\cdots<x_n$ 的 n 次插值多项式 $L_n(x)$，假定它满足条件

$$L_n(x_j) = y_j \qquad (j = 0,1,\cdots,n). \tag{2.6}$$

为了构造 $L_n(x)$，我们先定义 n 次插值基函数.

定义 1 若 n 次多项式 $l_j(x)(j=0,1,\cdots,n)$ 在 $n+1$ 个节点 $x_0<x_1<\cdots<x_n$ 上满足条件

$$l_j(x_k) = \begin{cases} 1, & k = j; \\ 0, & k \neq j. \end{cases} \quad (j,k = 0,1,\cdots,n) \tag{2.7}$$

就称这 $n+1$ 个 n 次多项式 $l_0(x), l_1(x), \cdots, l_n(x)$ 为节点 $x_0, x_1, \cdots, x_n$ 上的 ***n* 次插值基函数**.

对 $n=1$ 及 $n=2$ 时的情况前面已经讨论. 用类似的推导方法，可得到 n 次插值基函数为

$$l_k(x) = \frac{(x - x_0)\cdots(x - x_{k-1})(x - x_{k+1})\cdots(x - x_n)}{(x_k - x_0)\cdots(x_k - x_{k-1})(x_k - x_{k+1})\cdots(x_k - x_n)} \qquad (k = 0,1,\cdots,n). \tag{2.8}$$

显然它满足条件(2.7). 于是，满足条件(2.6)的插值多项式 $L_n(x)$ 可表示为

$$L_n(x) = \sum_{k=0}^{n} y_k l_k(x). \tag{2.9}$$

由 $l_k(x)$ 的定义，知

$$L_n(x_j)=\sum_{k=0}^{n}y_kl_k(x_j)=y_j \qquad (j=0,1,\cdots,n).$$

形如(2.9)的插值多项式 $L_n(x)$ 称为**拉格朗日(Lagrange)插值多项式**，而(2.3)与(2.5)是 $n=1$ 和 $n=2$ 的特殊情形.

若引入记号

$$\omega_{n+1}(x)=(x-x_0)(x-x_1)\cdots(x-x_n), \qquad (2.10)$$

容易求得

$$\omega_{n+1}'(x_k)=(x_k-x_0)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n).$$

于是公式(2.9)可改写成

$$L_n(x)=\sum_{k=0}^{n}y_k\frac{\omega_{n+1}(x)}{(x-x_k)\omega_{n+1}'(x_k)}. \qquad (2.11)$$

注意，n 次插值多项式 $L_n(x)$ 通常是次数为 n 的多项式，特殊情况下次数可能小于 n. 例如，通过三点 (x_0,y_0)，(x_1,y_1)，(x_2,y_2) 的二次插值多项式 $L_2(x)$，如果三点共线，则 $y=L_2(x)$ 就是一直线，而不是抛物线，这时 $L_2(x)$ 是一次多项式.

关于插值多项式存在唯一性有以下定理.

定理 1 在次数不超过 n 的多项式集合 H_n 中，满足条件(2.6)的插值多项式 $L_n(x)\in H_n$ 是存在唯一的.

证明 公式(2.11)所表示的 $L_n(x)$ 已证明了插值多项式的存在性，下面用反证法证明唯一性. 假定还有 $P(x)\in H_n$ 使 $P(x_i)=f(x_i)$，$i=0,1,\cdots,n$ 成立. 于是有 $L_n(x_i)-P(x_i)=0$ 对 $i=0,1,\cdots,n$ 成立，它表明多项式 $L_n(x)-P(x)\in H_n$ 有 $n+1$ 个零点 $x_0,x_1,\cdots,x_n$ 这与 n 次多项式只有 n 个零点的代数基本定理矛盾，故只能 $P(x)\equiv L_n(x)$. 证毕.

根据存在唯一性定理，若令 $f(x)=x^m$，$m=0,1,\cdots,n$ 可得

$$\sum_{k=0}^{n}x_k^ml_k(x)=x^m, \quad m=0,1,\cdots,n. \qquad (2.12)$$

若取 $m=0$，则

$$\sum_{k=0}^{n} l_k(x) = 1. \tag{2.13}$$

它可用来检验函数组$\{l_k(x), k=0,1,\cdots,n\}$的正确性.

2.2.3 插值余项与误差估计

若在$[a,b]$上用$L_n(x)$近似$f(x)$，则其截断误差为$R_n(x)=f(x)-L_n(x)$，也称为插值多项式的**余项**. 关于插值余项估计有以下定理.

定理2　设$f^{(n)}(x)$在$[a,b]$上连续，$f^{(n+1)}(x)$在(a,b)内存在，节点$a\leqslant x_0<x_1<\cdots<x_n\leqslant b$，$L_n(x)$是满足条件(2.6)的插值多项式，则对任何$x\in[a,b]$，插值余项

$$R_n(x) = f(x) - L_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x), \tag{2.14}$$

这里$\xi\in(a,b)$且依赖于x，$\omega_{n+1}(x)$是(2.10)所定义的.

证明　由给定条件知$R_n(x)$在节点$x_k(k=0,1,\cdots,n)$上为零，即$R_n(x_k)=0(k=0,1,\cdots,n)$，于是

$$R_n(x) = K(x)(x-x_0)(x-x_1)\cdots(x-x_n) = K(x)\omega_{n+1}(x), \tag{2.15}$$

其中$K(x)$是与x有关的待定函数.

现把x看成$[a,b]$上的一个固定点，作函数

$$\varphi(t) = f(t) - L_n(t) - K(x)(t-x_0)(t-x_1)\cdots(t-x_n),$$

根据插值条件及余项定义，可知$\varphi(t)$在点$x_0,x_1,\cdots,x_n$及x处均为零，故$\varphi(t)$在$[a,b]$上有$n+2$个零点，根据罗尔(Rolle)定理，$\varphi'(t)$在$\varphi(t)$的两个零点间至少有一个零点，故$\varphi'(t)$在$[a,b]$内至少有$n+1$个零点. 对$\varphi'(t)$再应用罗尔定理，可知$\varphi''(t)$在$[a,b]$内至少有n个零点. 依此类推，$\varphi^{(n+1)}(t)$在(a,b)内至少有一个零点，记为$\xi\in(a,b)$，使

$$\varphi^{(n+1)}(\xi) = f^{(n+1)}(\xi) - (n+1)!K(x) = 0,$$

于是

$$K(x)=\frac{f^{(n+1)}(\xi)}{(n+1)!},\xi\in(a,b),\text{且依赖于 } x.$$

将它代入(2.15)，就得到余项表达式(2.14). 证毕.

应当指出，余项表达式只有在 $f(x)$ 的高阶导数存在时才能应用. ξ 在 (a,b) 内的具体位置通常不可能给出，如果我们可以求出 $\max\limits_{a<x<b}|f^{(n+1)}(x)|=M_{n+1}$，那么插值多项式 $L_n(x)$ 逼近 $f(x)$ 的截断误差限是

$$|R_n(x)|\leqslant\frac{M_{n+1}}{(n+1)!}|\omega_{n+1}(x)|. \tag{2.16}$$

当 $n=1$ 时，线性插值余项为

$$R_1(x)=\frac{1}{2}f''(\xi)\omega_2(x)=\frac{1}{2}f''(\xi)(x-x_0)(x-x_1),$$

$$\xi\in[x_0,x_1]; \tag{2.17}$$

当 $n=2$ 时，抛物插值的余项为

$$R_2(x)=\frac{1}{6}f'''(\xi)(x-x_0)(x-x_1)(x-x_2),$$

$$\xi\in[x_0,x_2]. \tag{2.18}$$

例 1 已给 $\sin 0.32=0.314567$，$\sin 0.34=0.333487$，$\sin 0.36=0.352274$，用线性插值及抛物插值计算 $\sin 0.3367$ 的值并估计截断误差.

解 由题意取 $x_0=0.32$，$y_0=0.314567$，$x_1=0.34$，$y_1=0.333487$，$x_2=0.36$，$y_2=0.352274$.

用线性插值计算，取 $x_0=0.32$ 及 $x_1=0.34$，由公式(2.1)得

$$\sin 0.3367\approx L_1(0.3367)=y_0+\frac{y_1-y_0}{x_1-x_0}(0.3367-x_0)$$

$$=0.314567+\frac{0.01892}{0.02}\times 0.0167=0.330365.$$

其截断误差由(2.17)得

$$|R_1(x)| \leqslant \frac{M_2}{2}|(x-x_0)(x-x_1)|,$$

其中 $M_2=\max\limits_{x_0\leqslant x\leqslant x_1}|f''(x)|$，因 $f(x)=\sin x, f''(x)=-\sin x$. 可取 $M_2=\max\limits_{x_0\leqslant x\leqslant x_1}|\sin x|=\sin x_1\leqslant 0.3335$，于是

$$\begin{aligned}|R_1(0.3367)| &= |\sin 0.3367 - L_1(0.3367)| \\ &\leqslant \frac{1}{2}\times 0.3335\times 0.0167\times 0.0033\leqslant 0.92\times 10^{-5}.\end{aligned}$$

用抛物插值计算 sin0.3367 时，由公式(2.5)得

$$\begin{aligned}\sin 0.3367 &\approx y_0\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}+y_1\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} \\ &\quad + y_2\frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} \\ &= L_2(0.3367) \\ &= 0.314567\times\frac{0.7689\times 10^{-4}}{0.0008}+0.333487 \\ &\quad \times\frac{3.89\times 10^{-4}}{0.0004}+0.352274\times\frac{-0.5511\times 10^{-4}}{0.0008} \\ &= 0.330374.\end{aligned}$$

这个结果与6位有效数字的正弦函数表完全一样，这说明查表时用二次插值精度已相当高了. 其截断误差限由(2.18)得

$$|R_2(x)|\leqslant\frac{M_3}{6}|(x-x_0)(x-x_1)(x-x_2)|,$$

其中

$$M_3=\max_{x_0\leqslant x\leqslant x_2}|f'''(x)|=\cos x_0<0.828,$$

于是

$$\begin{aligned}|R_2(0.3367)| &= |\sin 0.3367 - L_2(0.3367)| \\ &\leqslant\frac{1}{6}\times 0.828\times 0.0167\times 0.033\times 0.0233 \\ &< 0.178\times 10^{-6}.\end{aligned}$$

2.3 均差与牛顿插值公式

2.3.1 均差及其性质

利用插值基函数很容易得到拉格朗日插值多项式，公式结构紧凑，在理论分析中甚为方便，但当插值节点增减时全部插值基函数 $l_k(x)(k=0,1,\cdots,n)$ 均要随之变化，整个公式也将发生变化，这在实际计算中是很不方便的，为了克服这一缺点，可把插值多项式表示为如下便于计算的形式

$$P_n(x)=a_0+a_1(x-x_0)+a_2(x-x_0)(x-x_1)+\cdots+a_n(x-x_0)\cdots(x-x_{n-1}), \tag{3.1}$$

其中 $a_0,a_1,\cdots,a_n$ 为待定系数，可由插值条件

$$P_n(x_j)=f_j \quad (j=0,1,\cdots,n)$$

确定.

当 $x=x_0$ 时，$P_n(x_0)=a_0=f_0$.

当 $x=x_1$ 时，$P_n(x_1)=a_0+a_1(x-x_0)=f_1$，推得

$$a_1=\frac{f_1-f_0}{x_1-x_0}.$$

当 $x=x_2$ 时，$P_n(x_2)=a_0+a_1(x_2-x_0)+a_2(x_2-x_0)(x_2-x_1)=f_2$，推得

$$a_2=\frac{\dfrac{f_2-f_0}{x_2-x_0}-\dfrac{f_1-f_0}{x_1-x_0}}{x_2-x_1}.$$

依此递推可得到 $a_3,\cdots,a_n$. 为写出系数 a_k 的一般表达式，先引进如下均差定义.

定义 2 称 $f[x_0,x_k]=\dfrac{f(x_k)-f(x_0)}{x_k-x_0}$ 为函数 $f(x)$ 关于点

x_0, x_k 的**一阶均差**. $f[x_0, x_1, x_k] = \dfrac{f[x_0, x_k] - f[x_0, x_1]}{x_k - x_1}$ 称为 $f(x)$ 的**二阶均差**. 一般地,称

$$f[x_0, x_1, \cdots, x_k] = \frac{f[x_0, \cdots, x_{k-2}, x_k] - f[x_0, x_1, \cdots, x_{k-1}]}{x_k - x_{k-1}} \tag{3.2}$$

为 $f(x)$ 的 **k 阶均差**(均差也称为**差商**).

均差有如下的基本性质:

1° k 阶均差可表为函数值 $f(x_0), \cdots, f(x_k)$ 的线性组合,即

$$f[x_0, \cdots, x_k] = \sum_{j=0}^{k} \frac{f(x_j)}{(x_j - x_0)\cdots(x_j - x_{j-1})(x_j - x_{j+1})\cdots(x_j - x_k)}. \tag{3.3}$$

这个性质可用归纳法证明. 这性质也表明均差与节点的排列次序无关,称为均差的对称性. 即

$$f[x_0, \cdots, x_k] = f[x_1, x_0, x_2, \cdots, x_k] = \cdots = f[x_1, \cdots, x_k, x_0].$$

2° 由性质 1°及(3.2)可得

$$f[x_0, \cdots, x_k] = \frac{f[x_1, \cdots, x_k] - f[x_0, \cdots, x_{k-1}]}{x_k - x_0}. \tag{3.4}$$

3° 若 $f(x)$ 在 $[a, b]$ 上存在 n 阶导数,且节点 $x_0, \cdots, x_n \in [a, b]$,则 n 阶均差与导数关系如下:

$$f[x_0, \cdots, x_n] = \frac{f^{(n)}(\xi)}{n!}, \qquad \xi \in [a, b]. \tag{3.5}$$

这公式可直接用罗尔定理证明.

均差的其他性质还可见习题. 均差计算可列均差表如下(表 2-1).

表 2-1

x_k	$f(x_k)$	一阶均差	二阶均差	三阶均差	四阶均差
x_0	$\underline{f(x_0)}$				
x_1	$f(x_1)$	$\underline{f[x_0,x_1]}$			
x_2	$f(x_2)$	$f[x_1,x_2]$	$\underline{f[x_0,x_1,x_2]}$		
x_3	$f(x_3)$	$f[x_2,x_3]$	$f[x_1,x_2,x_3]$	$\underline{f[x_0,x_1,x_2,x_3]}$	
x_4	$f(x_4)$	$f[x_3,x_4]$	$f[x_2,x_3,x_4]$	$f[x_1,x_2,x_3,x_4]$	$\underline{f[x_0,x_1,x_2,x_3,x_4]}$
⋮	⋮	⋮	⋮	⋮	⋮

2.3.2 牛顿插值公式

根据均差定义，把 x 看成$[a,b]$上一点，可得

$$f(x) = f(x_0) + f[x,x_0](x-x_0),$$
$$f[x,x_0] = f[x_0,x_1] + f[x,x_0,x_1](x-x_1),$$
$$\cdots$$
$$f[x,x_0,\cdots,x_{n-1}] = f[x_0,x_1,\cdots,x_n] + f[x,x_0,\cdots,x_n](x-x_n).$$

只要把后一式代入前一式，就得到

$$\begin{aligned} f(x) =& f(x_0) + f[x_0,x_1](x-x_0) \\ &+ f[x_0,x_1,x_2](x-x_0)(x-x_1) + \cdots \\ &+ f[x_0,x_1,\cdots,x_n](x-x_0)\cdots(x-x_{n-1}) \\ &+ f[x,x_0,\cdots,x_n]\omega_{n+1}(x) = N_n(x) + R_n(x), \end{aligned}$$

其中

$$\begin{aligned} N_n(x) =& f(x_0) + f[x_0,x_1](x-x_0) \\ &+ f[x_0,x_1,x_2](x-x_0)(x-x_1) + \cdots \\ &+ f[x_0,\cdots,x_n](x-x_0)\cdots(x-x_{n-1}), \end{aligned} \tag{3.6}$$

$$R_n(x) = f(x) - N_n(x) = f[x,x_0,\cdots,x_n]\omega_{n+1}(x), \tag{3.7}$$

$\omega_{n+1}(x)$是由(2.10)定义的.

由(3.6)确定的多项式 $N_n(x)$显然满足插值条件，且次数不

超过 n,它就是形如(3.1)的多项式,其系数为

$$a_k = f[x_0,\cdots,x_k] \quad (k=0,1,\cdots,n).$$

我们称 $N_n(x)$ 为**牛顿(Newton)均差插值多项式**. 系数 a_k 就是均差表 2-1 中加横线的各阶均差,它比拉格朗日插值计算量省,且便于程序设计.

(3.7)为插值余项,由插值多项式唯一性知,它与(2.14)是等价的,事实上,利用均差与导数关系式(3.5)可由(3.7)推出(2.14). 但(3.7)更有一般性,它对 f 是由离散点给出的情形或 f 导数不存在时均适用.

例 2 给出 $f(x)$ 的函数表(见表 2-2),求 4 次牛顿插值多项式,并由此计算 $f(0.596)$ 的近似值.

首先根据给定函数表造出均差表.

表 2-2

0.40	0.41075					
0.55	0.57815	1.11600				
0.65	0.69675	1.18600	0.28000			
0.80	0.88811	1.27573	0.35893	0.19733		
0.90	1.02652	1.38410	0.43348	0.21300	0.03134	
1.05	1.25382	1.51533	0.52493	0.22863	0.03126	−0.00012

从均差表看到 4 阶均差近似常数. 故取 4 次插值多项式 $N_4(x)$ 做近似即可.

$$\begin{aligned} N_4(x) =& 0.41075+1.116(x-0.4)+0.28(x-0.4)(x-0.55) \\ &+0.19733(x-0.4)(x-0.55)(x-0.65) \\ &+0.03134(x-0.4)(x-0.55)(x-0.65)(x-0.8), \end{aligned}$$

于是

$$f(0.596) \approx N_4(0.596) = 0.63192,$$

截断误差

$$|R_4(x)| \approx |f[x_0,\cdots,x_5]\omega_5(0.596)| \leqslant 3.63\times 10^{-9}.$$

这说明截断误差很小,可忽略不计.

此例的截断误差估计中,5 阶均差 $f[x,x_0,\cdots,x_4]$用 $f[x_0,x_1,\cdots,x_5]=-0.00012$ 近似. 另一种方法是取 $x=0.596$,由 $f(0.596)\approx 0.63192$,可求得 $f[x,x_0,\cdots,x_4]$的近似值,从而可求得$|R_4(x)|$的近似.

2.4 差分与等距节点插值

上面讨论了节点任意分布的插值公式,但实际应用时经常遇到等距节点的情形,这时插值公式可以进一步简化,计算也简单得多. 为了得到等距节点的插值公式,我们先介绍差分的概念.

2.4.1 差分及其性质

设函数 $y=f(x)$在等距节点 $x_k=x_0+kh(k=0,1,\cdots,n)$上的值 $f_k=f(x_k)$为已知,这里 h 为常数,称为**步长**.

定义 3 记号

$$\Delta f_k = f_{k+1} - f_k, \tag{4.1}$$

$$\nabla f_k = f_k - f_{k-1}, \tag{4.2}$$

$$\delta f_k = f(x_k + h/2) - f(x_k - h/2) = f_{k+\frac{1}{2}} - f_{k-\frac{1}{2}} \tag{4.3}$$

分别称为 $f(x)$在 x_k 处以 h 为步长的**向前差分**,**向后差分**及**中心差分**. 符号 Δ,∇,δ 分别称为**向前差分算子**,**向后差分算子**及**中心差分算子**.

利用一阶差分可定义二阶差分为

$$\Delta^2 f_k = \Delta f_{k+1} - \Delta f_k = f_{k+2} - 2f_{k+1} + f_k.$$

一般地可定义 **m 阶差分**为

$$\Delta^m f_k = \Delta^{m-1} f_{k+1} - \Delta^{m-1} f_k;$$

$$\nabla^m f_k = \nabla^{m-1} f_k - \nabla^{m-1} f_{k-1}.$$

因中心差心 δf_k 用到 $f_{k+\frac{1}{2}}$及 $f_{k-\frac{1}{2}}$这两个值,实际上不是函数表上的值,如果用函数表上的值,一阶中心差分应写成

$$\delta f_{k+\frac{1}{2}} = f_{k+1} - f_k,\ \delta f_{k-\frac{1}{2}} = f_k - f_{k-1},$$

二阶中心差分为

$$\delta^2 f_k = \delta f_{k+\frac{1}{2}} - \delta f_{k-\frac{1}{2}},$$

等等.

除了已引入的差分算子外，常用算子符号还有**不变算子** I 及**移位算子** E，定义如下：

$$If_k = f_k,\quad Ef_k = f_{k+1},$$

于是，由 $\Delta f_k = f_{k+1} - f_k = Ef_k - If_k = (E-I)f_k$，可得

$$\Delta = E - I,$$

同理可得

$$\nabla = I - E^{-1}, \qquad \delta = E^{\frac{1}{2}} - E^{-\frac{1}{2}}.$$

由差分定义并应用算子符号运算可得下列基本性质.

性质 1 各阶差分均可用函数值表示. 例如

$$\begin{aligned}\Delta^n f_k = (E-I)^n f_k &= \sum_{j=0}^{n} (-1)^j \binom{n}{j} E^{n-j} f_k \\ &= \sum_{j=0}^{n} (-1)^j \binom{n}{j} f_{n+k-j},\end{aligned} \tag{4.4}$$

$$\begin{aligned}\nabla^n f_k = (I-E^{-1})^n f_k &= \sum_{j=0}^{n} (-1)^{n-j} \binom{n}{j} E^{j-n} f_k \\ &= \sum_{j=0}^{n} (-1)^{n-j} \binom{n}{j} f_{k+j-n},\end{aligned} \tag{4.5}$$

其中 $\binom{n}{j} = \dfrac{n(n-1)\cdots(n-j+1)}{j!}$ 为二项式展开系数.

性质 2 可用各阶差分表示函数值. 例如，可用向前差分表示 f_{n+k}，因为

$$f_{n+k} = E^n f_k = (I+\Delta)^n f_k = \left[\sum_{j=0}^{n} \binom{n}{j} \Delta^j\right] f_k,$$

于是

$$f_{n+k} = \sum_{j=0}^{n} \binom{n}{j} \Delta^j f_k. \tag{4.6}$$

性质 3 均差与差分有密切关系，例如，对向前差分，由定义

$$f[x_k, x_{k+1}] = \frac{f_{k+1} - f_k}{x_{k+1} - x_k} = \frac{\Delta f_k}{h},$$

$$f[x_k, x_{k+1}, x_{k+2}] = \frac{f[x_{k+1}, x_{k+2}] - f[x_k, x_{k+1}]}{x_{k+2} - x_k}$$

$$= \frac{1}{2h^2}\Delta^2 f_k,$$

一般地有

$$f[x_k, \cdots, x_{k+m}] = \frac{1}{m!}\frac{1}{h^m}\Delta^m f_k \quad (m = 1, 2, \cdots, n). \tag{4.7}$$

同理,对向后差分有

$$f[x_k, x_{k-1}, \cdots, x_{k-m}] = \frac{1}{m!}\frac{1}{h^m}\nabla^m f_k. \tag{4.8}$$

利用(4.7)及(3.5)又可得到

$$\Delta^n f_k = h^n f^{(n)}(\xi), \tag{4.9}$$

其中 $\xi \in (x_k, x_{k+n})$,这就是差分与导数的关系. 差分的其他性质可参看本章习题.

计算差分可列差分表(见表 2-3),表中 Δ 为向前差分,∇ 为向后差分.

表 2-3

f_k	$\Delta(\nabla)$	$\Delta^2(\nabla^2)$	$\Delta^3(\nabla^3)$	$\Delta^4(\nabla^4)$	$\cdots$
f_0					
	$\Delta f_0(\nabla f_1)$				
f_1		$\Delta^2 f_0(\nabla^2 f_2)$			
	$\Delta f_1(\nabla f_2)$		$\Delta^3 f_0(\nabla^3 f_3)$		
f_2		$\Delta^2 f_1(\nabla^2 f_3)$		$\Delta^4 f_0(\nabla^4 f_4)$	
	$\Delta f_2(\nabla f_3)$		$\Delta^3 f_1(\nabla^3 f_4)$		$\vdots$
f_3		$\Delta^2 f_2(\nabla^2 f_4)$		$\vdots$	
	$\Delta f_3(\nabla f_4)$		$\vdots$		
f_4		$\vdots$			
	$\vdots$				
$\vdots$					

2.4.2 等距节点插值公式

将牛顿均差插值多项式(3.6)中各阶均差用相应差分代替,就可得到各种形式的等距节点插值公式. 这里只推导常用的前插与后插公式.

如果节点 $x_k=x_0+kh(k=0,1,\cdots,n)$,要计算 x_0 附近点 x 的函数 $f(x)$的值,可令 $x=x_0+th,0\leqslant t\leqslant 1$,于是

$$\omega_{k+1}(x)=\prod_{j=0}^{k}(x-x_j)=t(t-1)\cdots(t-k)h^{k+1}.$$

将此式及(4.7)代入(3.6),则得

$$N_n(x_0+th)=f_0+t\Delta f_0+\frac{t(t-1)}{2!}\Delta^2 f_0+\cdots+\frac{t(t-1)\cdots(t-n+1)}{n!}\Delta^n f_0, \tag{4.10}$$

称为**牛顿前插公式**,其余项由(2.14)得

$$R_n(x)=\frac{t(t-1)\cdots(t-n)}{(n+1)!}h^{n+1}f^{(n+1)}(\xi),\ \xi\in(x_0,x_n). \tag{4.11}$$

如果要求函数表示 x_n 附近的函数值 $f(x)$,此时应用牛顿插值公式(3.6),插值点应按 $x_n,x_{n-1},\cdots,x_0$ 的次序排列,有

$$\begin{aligned}N_n(x)=&f(x_n)+f[x_n,x_{n-1}](x-x_n)\\&+f[x_n,x_{n-1},x_{n-2}](x-x_n)(x-x_{n-1})+\cdots\\&+f[x_n,x_{n-1},\cdots,x_0](x-x_n)\cdots(x-x_1).\end{aligned}$$

作变换 $x=x_n+th(-1\leqslant t\leqslant 0)$,并利用公式(4.8),代入上式得

$$N_n(x_n+th)=f_n+t\nabla f_n+\frac{t(t+1)}{2!}\nabla^2 f_n+\cdots+\frac{t(t+1)\cdots(t+n-1)}{n!}\nabla^n f_n, \tag{4.12}$$

称其为**牛顿后插公式**,其余项

$$
\begin{aligned}
R_n(x) &= f(x) - N_n(x_n + th) \\
&= \frac{t(t+1)\cdots(t+n)h^{n+1}f^{(n+1)}(\xi)}{(n+1)!},
\end{aligned}
\qquad (4.13)
$$

其中 $\xi \in (x_0, x_n)$.

通常求开头部分插值点附近函数值时使用牛顿前插公式,求插值节点末尾附近函数值时使用牛顿后插公式. 如果用相同节点进行插值,则向前向后两种公式只是形式上差别,其计算结果是相同的.

例 3 给出 $f(x)=\cos x$ 在 $x_k=kh, k=0,1,\cdots,6, h=0.1$ 处的函数值,试用 4 次等距节点插值公式计算 $f(0.048)$ 及 $f(0.566)$ 的近似值并估计误差.

解 先构造差分表. 用牛顿向前插值公式(4.10)计算 $f(0.048)$ 的近似值,取 $x=0.048, h=0.1, t=\dfrac{x-0}{h}=0.48$,用表 2-4 上半部差分,得

表 2-4

$f(x_k)$	$\Delta f(\nabla f)$	$\Delta^2 f(\nabla^2 f)$	$\Delta^3 f(\nabla^3 f)$	$\Delta^4 f(\nabla^4 f)$	$\Delta^5 f(\nabla^5 f)$
1.00000					
	−0.00500				
0.99500		−0.00993			
	−0.01493		0.00013		
0.98007		−0.00980		0.00012	
	−0.02473		0.00025		−0.00002
0.95534		−0.00955		0.00010	
	−0.03428		0.00035		−0.00001
0.92106		−0.00920		0.00009	

续表

$f(x_k)$	$\Delta f(\nabla f)$	$\Delta^2 f(\nabla^2 f)$	$\Delta^3 f(\nabla^3 f)$	$\Delta^4 f(\nabla^4 f)$	$\Delta^5 f(\nabla^5 f)$
	−0.04348		0.00044		
0.87758		−0.00876			
	−0.05224				
0.82534					

$$
\begin{aligned}
N_4(0.048) = & 1.00000 + 0.48 \times (-0.00500) \\
& + \frac{(0.48)(0.48-1)}{2}(-0.00993) \\
& + \frac{1}{3!}(0.48)(0.48-1)(0.48-2)(0.00013) \\
& + \frac{1}{4!}(0.48)(0.48-1)(0.48-2)(0.48-3)(0.00012) \\
= & 0.99885 \approx \cos 0.048,
\end{aligned}
$$

误差估计由(4.11)可得

$$
\begin{aligned}
|R_4(0.048)| & \leqslant \frac{M_5}{5!} |t(t-1)(t-2)(t-3)(t-4)| h^5 \\
& \leqslant 1.5845 \times 10^{-7},
\end{aligned}
$$

其中 $M_5 = |\sin 0.6| \leqslant 0.565$.

计算 $f(0.566)$. 可用牛顿向后插值公式(4.12), $x=0.566$, $x_6=0.6$, $t=\dfrac{x-x_6}{h}=-0.34$,用差分表 2-4 中下半部差分,得

$$
\begin{aligned}
N_4(0.566) = & 0.82534 - 0.34\left[-0.05224 + (0.66)\left(\frac{-0.00876}{2}\right.\right. \\
& \left.\left. + (1.66)\left(\frac{0.00044}{6} + 2.66 \times \frac{0.00009}{24}\right)\right)\right] \\
= & 0.84405.
\end{aligned}
$$

于是 $\cos 0.566 \approx 0.84405$,误差估计由(4.13)得

$$|R_4(0.566)| \leqslant \frac{M_5}{5!}|t(t+1)(t+2)(t+3)(t+4)|h^5$$
$$\leqslant 1.7064 \times 10^{-7},$$

其中 $M_5=0.565$.

2.5 埃尔米特插值

不少实际的插值问题不但要求在节点上函数值相等，而且还要求对应的导数值也相等，甚至要求高阶导数也相等，满足这种要求的插值多项式就是**埃尔米特(Hermite)插值多项式**. 下面只讨论函数值与导数值个数相等的情况. 设在节点 $a\leqslant x_0<x_1<\cdots<x_n\leqslant b$ 上，$y_j=f(x_j)$，$m_j=f'(x_j)$ $(j=0,1,\cdots,n)$，要求插值多项式 $H(x)$，满足条件

$$H(x_j)=y_j, H'(x_j)=m_j \quad (j=0,1,\cdots,n). \tag{5.1}$$

这里给出了 $2n+2$ 个条件，可唯一确定一个次数不超过 $2n+1$ 的多项式 $H_{2n+1}(x)=H(x)$，其形式为

$$H_{2n+1}(x)=a_0+a_1x+\cdots+a_{2n+1}x^{2n+1},$$

如根据条件(5.1)来确定 $2n+2$ 个系数 $a_0,a_1,\cdots,a_{2n+1}$，显然非常复杂，因此，我们仍采用求拉格朗日插值多项式的基函数方法. 先求插值基函数 $\alpha_j(x)$ 及 $\beta_j(x)$ $(j=0,1,\cdots,n)$，共有 $2n+2$ 个，每一个基函数都是 $2n+1$ 次多项式，且满足条件

$$\left.\begin{aligned}&\alpha_j(x_k)=\delta_{jk}=\begin{cases}0, & j\neq k,\\ 1, & j=k,\end{cases}\quad \alpha_j'(x_k)=0;\\ &\beta_j(x_k)=0,\ \beta_j'(x_k)=\delta_{jk}(j,k=0,1,\cdots,n).\end{aligned}\right\} \tag{5.2}$$

于是满足条件(5.1)的插值多项式 $H(x)=H_{2n+1}(x)$ 可写成用插值基函数表示的形式

$$H_{2n+1}(x)=\sum_{j=0}^{n}[y_j\alpha_j(x)+m_j\beta_j(x)]. \tag{5.3}$$

由条件(5.2)，显然有 $H_{2n+1}(x_k)=y_k, H'_{2n+1}(x_k)=m_k, (k=0, 1,\cdots,n)$. 下面的问题就是求满足条件(5.2)的基函数 $\alpha_j(x)$ 及 $\beta_j(x)$. 为此，可利用拉格朗日插值基函数 $l_j(x)$. 令

$$\alpha_j(x) = (ax+b)l_j^2(x),$$

其中 $l_j(x)$ 是(2.8)所表示的基函数. 由条件(5.2)有

$$\alpha_j(x_j) = (ax_j+b)l_j^2(x_j) = 1,$$

$$\alpha_j'(x_j) = l_j(x_j)[al_j(x_j)+2(ax_j+b)l_j'(x_j)] = 0,$$

整理得

$$\begin{cases} ax_j+b=1; \\ a+2l_j'(x_j)=0. \end{cases}$$

解出

$$a=-2l_j'(x_j),\quad b=1+2x_jl_j'(x_j).$$

由于

$$l_j(x) = \frac{(x-x_0)\cdots(x-x_{j-1})(x-x_{j+1})\cdots(x-x_n)}{(x_j-x_0)\cdots(x_j-x_{j-1})(x_j-x_{j+1})\cdots(x_j-x_n)},$$

利用两端取对数再求导，得

$$l_j'(x_j) = \sum_{\substack{k=0\\k\neq j}}^{n} \frac{1}{x_j-x_k},$$

于是

$$\alpha_j(x) = \left(1-2(x-x_j)\sum_{\substack{k=0\\k\neq j}}^{n}\frac{1}{x_j-x_k}\right)l_j^2(x). \tag{5.4}$$

同理，可得

$$\beta_j(x) = (x-x_j)l_j^2(x). \tag{5.5}$$

还可证明满足条件(5.1)的插值多项式是唯一的. 用反证法，假设 $H_{2n+1}(x)$ 及 $\overline{H}_{2n+1}(x)$ 均满足条件(5.1)，于是

$$\varphi(x) = H_{2n+1}(x) - \overline{H}_{2n+1}(x)$$

在每个节点 x_k 上均有二重根，即 $\varphi(x)$ 有 $2n+2$ 重根. 但 $\varphi(x)$ 是不高于 $2n+1$ 次的多项式，故 $\varphi(x)\equiv 0$. 唯一性得证.

仿照拉格朗日插值余项的证明方法，若 $f(x)$ 在 (a,b) 内的 $2n+2$ 阶导数存在，则其插值余项

$$R(x) = f(x) - H_{2n+1}(x) = \frac{f^{(2n+2)}(\xi)}{(2n+2)!}\omega_{n+1}^2(x), \quad (5.6)$$

其中 $\xi \in (a,b)$ 且与 x 有关. 具体证明请读者自行完成.

作为带导数插值多项式(5.3)的重要特例是 $n=1$ 的情形. 这时可取节点为 x_k 及 x_{k+1}，插值多项式为 $H_3(x)$，满足条件

$$\left.\begin{aligned} &H_3(x_k) = y_k, \quad H_3(x_{k+1}) = y_{k+1}; \\ &H_3'(x_k) = m_k, \quad H_3'(x_{k+1}) = m_{k+1}. \end{aligned}\right\} \quad (5.7)$$

相应的插值基函数为 $\alpha_k(x), \alpha_{k+1}(x), \beta_k(x), \beta_{k+1}(x)$，它们满足条件

$$\begin{aligned} &\alpha_k(x_k) = 1,\ \alpha_k(x_{k+1}) = 0, \\ &\alpha_k'(x_k) = \alpha_k'(x_{k+1}) = 0, \\ &\alpha_{k+1}(x_k) = 0,\ \alpha_{k+1}(x_{k+1}) = 1, \\ &\alpha_{k+1}'(x_k) = \alpha_{k+1}'(x_{k+1}) = 0; \\ &\beta_k(x_k) = \beta_k(x_{k+1}) = 0, \\ &\beta_k'(x_k) = 1, \beta_k'(x_{k+1}) = 0, \\ &\beta_{k+1}(x_k) = \beta_{k+1}(x_{k+1}) = 0, \\ &\beta_{k+1}'(x_k) = 0,\ \beta_{k+1}'(x_{k+1}) = 1. \end{aligned}$$

根据(5.4)及(5.5)的一般表达式，可得到

$$\left.\begin{aligned} \alpha_k(x) &= \left(1 + 2\frac{x - x_k}{x_{k+1} - x_k}\right)\left(\frac{x - x_{k+1}}{x_k - x_{k+1}}\right)^2, \\ \alpha_{k+1}(x) &= \left(1 + 2\frac{x - x_{k+1}}{x_k - x_{k+1}}\right)\left(\frac{x - x_k}{x_{k+1} - x_k}\right)^2. \end{aligned}\right\} \quad (5.8)$$

$$\left.\begin{aligned} \beta_k(x) &= (x - x_k)\left(\frac{x - x_{k+1}}{x_k - x_{k+1}}\right)^2, \\ \beta_{k+1}(x) &= (x - x_{k+1})\left(\frac{x - x_k}{x_{k+1} - x_k}\right)^2. \end{aligned}\right\} \quad (5.9)$$

于是满足条件(5.7)的插值多项式是

$$H_3(x)=y_k\alpha_k(x)+y_{k+1}\alpha_{k+1}(x)+m_k\beta_k(x)+m_{k+1}\beta_{k+1}(x), \tag{5.10}$$

其余项 $R_3(x)=f(x)-H_3(x)$，由(5.6)得

$$R_3(x)=\frac{1}{4!}f^{(4)}(\xi)(x-x_k)^2(x-x_{k+1})^2,\xi\in(x_k,x_{k+1}).$$

例4 求满足 $P(x_j)=f(x_j)(j=0,1,2)$ 及 $P'(x_1)=f'(x_1)$ 的插值多项式及其余项表达式.

由给定条件，可确定次数不超过3的插值多项式. 由于此多项式通过点 $(x_0,f(x_0))$，$(x_1,f(x_1))$ 及 $(x_2,f(x_2))$，故其形式为

$$\begin{aligned}P(x)=&f(x_0)+f[x_0,x_1](x-x_0)\\&+f[x_0,x_1,x_2](x-x_0)(x-x_1)\\&+A(x-x_0)(x-x_1)(x-x_2),\end{aligned}$$

其中 A 为待定常数，可由条件 $P'(x_1)=f'(x_1)$ 确定，通过计算可得

$$A=\frac{f'(x_1)-f[x_0,x_1]-(x_1-x_0)f[x_0,x_1,x_2]}{(x_1-x_0)(x_1-x_2)}.$$

为了求出余项 $R(x)=f(x)-P(x)$ 的表达式，可设

$$R(x)=f(x)-P(x)=k(x)(x-x_0)(x-x_1)^2(x-x_2),$$

其中 $k(x)$ 为待定函数. 构造

$$\varphi(t)=f(t)-P(t)-k(x)(t-x_0)(t-x_1)^2(t-x_2).$$

显然 $\varphi(x_j)=0(j=0,1,2)$. 且 $\varphi'(x_1)=0$，$\varphi(x)=0$，故 $\varphi(t)$ 在 (a,b) 内有5个零点(二重根算两个). 反复应用罗尔定理，得 $\varphi^{(4)}(t)$ 在 (a,b) 内至少有一个零点 ξ，故

$$\varphi^{(4)}(\xi)=f^{(4)}(\xi)-4!k(x)=0,$$

于是

$$k(x)=\frac{1}{4!}f^{(4)}(\xi),$$

余项表达式为

$$R(x)=\frac{1}{4!}f^{(4)}(\xi)(x-x_0)(x-x_1)^2(x-x_2),\quad (5.11)$$

式中 ξ 位于 x_0,x_1,x_2 和 x 所界定的范围内.

2.6 分段低次插值

2.6.1 高次插值的病态性质

上面我们根据区间$[a,b]$上给出的节点做插值多项式 $L_n(x)$ 近似 $f(x)$，一般总认为 $L_n(x)$的次数 n 越高逼近 $f(x)$的精度越好，但实际上并非如此. 这是因为对任意的插值节点，当 $n\to\infty$时，$L_n(x)$不一定收敛到 $f(x)$. 20 世纪初龙格(Runge)就给出了一个等距节点插值多项式 $L_n(x)$不收敛到 $f(x)$的例子. 他给出的函数为 $f(x)=1/(1+x^2)$，它在$[-5,5]$上各阶导数均存在. 在$[-5,5]$上取 $n+1$ 个等距节点 $x_k=-5+10\frac{k}{n}(k=0,1,\cdots,n)$所构造的拉格朗日插值多项式为

$$L_n(x)=\sum_{j=0}^{n}\frac{1}{1+x_j^2}\frac{\omega_{n+1}(x)}{(x-x_j)\omega_{n+1}'(x_j)}.$$

令 $x_{n-1/2}=\frac{1}{2}(x_{n-1}+x_n)$，则 $x_{n-1/2}=5-\frac{5}{n}$，表 2-5 列出了 $n=2,4,\cdots,20$ 的 $L_n(x_{n-1/2})$的计算结果及在 $x_{n-1/2}$上的误差$R(x_{n-1/2})$. 可以看出，随 n 的增加，$R(x_{n-1/2})$的绝对值几乎成倍地增加. 这说明当 $n\to\infty$时 L_n 在$[-5,5]$上不收敛. Runge 证明了，存在一个常数 $c\approx3.63$，使得当$|x|\leqslant c$ 时，$\lim\limits_{n\to\infty}L_n(x)=f(x)$，而当$|x|>c$ 时$\{L_n(x)\}$发散.

下面取 $n=10$，根据计算画出 $y=L_{10}(x)$及 $y=1/(1+x^2)$在$[-5,5]$上的图形，见图 2-5.

表　2-5

n	$f(x_{n-1/2})$	$L_n(x_{n-1/2})$	$R(x_{n-1/2})$
2	0.137931	0.759615	−0.621684
4	0.066390	−0.356826	0.423216
6	0.054463	0.607879	−0.553416
8	0.049651	−0.831017	0.880668
10	0.047059	1.578721	−1.531662
12	0.045440	−2.755000	2.800440
14	0.044334	5.332743	−5.288409
16	0.043530	−10.173867	10.217397
18	0.042920	20.123671	−20.080751
20	0.042440	−39.952449	39.994889

从图上看到，在 $x=\pm5$ 附近 $L_{10}(x)$ 与 $f(x)=1/(1+x^2)$ 偏离很远，例如 $L_{10}(4.8)=1.80438$，$f(4.8)=0.04160$. 这说明用高次插值多项式 $L_n(x)$ 近似 $f(x)$ 效果并不好，因而通常不用高次插值，而用分段低次插值. 从本例看到，如果我们把 $y=1/(1+x^2)$ 在节点 $x=0,\pm1,\pm2,\pm3,\pm4,\pm5$ 处用折线连起来显然比 $L_{10}(x)$ 逼近 $f(x)$ 好得多. 这正是我们下面要讨论的分段低次插值的出发点.

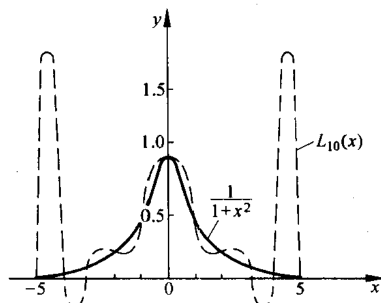


图　2-5

2.6.2 分段线性插值

所谓分段线性插值就是通过插值点用折线段连接起来逼近 $f(x)$. 设已知节点 $a=x_0<x_1<\cdots<x_n=b$ 上的函数值 $f_0,f_1,\cdots,f_n$,记 $h_k=x_{k+1}-x_k$, $h=\max\limits_k h_k$,求一折线函数 $I_h(x)$满足:

1° 记 $I_h(x)\in C[a,b]$,

2° $I_h(x_k)=f_k \quad (k=0,1,\cdots,n)$,

3° $I_h(x)$在每个小区间$[x_k,x_{k+1}]$上是线性函数。

则称 $I_h(x)$为**分段线性插值函数**.

由定义可知 $I_h(x)$在每个小区间$[x_k,x_{k+1}]$上可表示为

$$I_h(x)=\frac{x-x_{k+1}}{x_k-x_{k+1}}f_k+\frac{x-x_k}{x_{k+1}-x_k}f_{k+1} \quad (x_k\leqslant x\leqslant x_{k+1}). \tag{6.1}$$

若用插值基函数表示,则在整个区间$[a,b]$上 $I_h(x)$为

$$I_h(x)=\sum_{j=0}^{n}f_jl_j(x), \tag{6.2}$$

其中基函数 $l_j(x)$满足条件 $l_j(x_k)=\delta_{jk}(j,k=0,1,\cdots,n)$,其形式是

$$l_j(x)=\begin{cases}\dfrac{x-x_{j-1}}{x_j-x_{j-1}}, & x_{j-1}\leqslant x\leqslant x_j(j=0\text{ 略去});\\ \dfrac{x-x_{j+1}}{x_j-x_{j+1}}, & x_j\leqslant x\leqslant x_{j+1}(j=n\text{ 略去});\\ 0, & x\in[a,b],x\notin[x_{j-1},x_{j+1}].\end{cases} \tag{6.3}$$

分段线性插值的误差估计可利用插值余项(2.17)得到

$$\max_{x_k\leqslant x\leqslant x_{k+1}}|f(x)-I_h(x)|\leqslant\frac{M_2}{2}\max_{x_k\leqslant x\leqslant x_{k+1}}|(x-x_k)(x-x_{k+1})|$$

或

$$\max_{a\leqslant x\leqslant b}|f(x)-I_h(x)|\leqslant\frac{M_2}{8}h^2, \tag{6.4}$$

其中 $M_2 = \max\limits_{a \leqslant x \leqslant b} |f''(x)|$.

分段线性插值基函数 $l_j(x)$ 只在 x_j 附近不为零，在其他地方均为零，这种性质称为**局部非零性质**. 当 $x \in [x_k, x_{k+1}]$ 时

$$1 = \sum_{j=0}^{n} l_j(x) = l_k(x) + l_{k+1}(x),$$

故

$$f(x) = [l_k(x) + l_{k+1}(x)]f(x).$$

另一方面，这时

$$I_h(x) = f_k l_k(x) + f_{k+1} l_{k+1}(x).$$

现在证明 $\lim\limits_{h \to 0} I_h(x) = f(x)$. 考虑

$$\begin{aligned} |f(x) - I_h(x)| &\leqslant l_k(x)|f(x) - f_k| \\ &\quad + l_{k+1}(x)|f(x) - f_{k+1}| \\ &\leqslant [l_k(x) + l_{k+1}(x)]\omega(h_k) = \omega(h_k) \leqslant \omega(h). \end{aligned}$$

这里 $\omega(h)$ 是函数 $f(x)$ 在区间 $[a,b]$ 上的**连续模**，即对任意两点 $x', x'' \in [a,b]$，只要 $|x' - x''| \leqslant h$，就有

$$|f(x') - f(x'')| \leqslant \omega(h)$$

称 $\omega(h)$ 为 $f(x)$ 在 $[a,b]$ 上的连续模，当 $f(x) \in C[a,b]$ 时，就有 $\lim\limits_{h \to 0} \omega(h) = 0$.

由前式可知，当 $x \in [a,b]$ 时有

$$\max_{a \leqslant x \leqslant b} |f(x) - I_h(x)| \leqslant \omega(h).$$

因此，只要 $f(x) \in C[a,b]$，就有

$$\lim_{h \to 0} I_h(x) = f(x)$$

在 $[a,b]$ 上一致成立，故 $I_h(x)$ 在 $[a,b]$ 上一致收敛到 $f(x)$.

2.6.3　分段三次埃尔米特插值

分段线性插值函数 $I_h(x)$ 的导数是间断的，若在节点 $x_k(k=0,1,\cdots,n)$ 上除已知函数值 f_k 外还给出导数值 $f'_k = m_k(k=0,1,\cdots,n)$，这样就可构造一个导数连续的分段插值函数 $I_h(x)$，它

满足条件：

1. $I_h(x)\in C^1[a,b]$($C^1[a,b]$代表区间$[a,b]$上一阶导数连续的函数集合)，

2. $I_h(x_k)=f_k, I_h'(x_k)=f_k'(k=0,1,\cdots,n)$，

3. $I_h(x)$在每个小区间$[x_k,x_{k+1}]$上是三次多项式.

根据两点三次插值多项式(5.10). 可知，$I_h(x)$在区间$[x_k,x_{k+1}]$上的表达式为

$$
\begin{aligned}
I_h(x) = &\left(\frac{x-x_{k+1}}{x_k-x_{k+1}}\right)^2\left(1+2\frac{x-x_k}{x_{k+1}-x_k}\right)f_k+\left(\frac{x-x_k}{x_{k+1}-x_k}\right)^2 \\
&\cdot\left(1+2\frac{x-x_{k+1}}{x_k-x_{k+1}}\right)f_{k+1}+\left(\frac{x-x_{k+1}}{x_k-x_{k+1}}\right)^2(x-x_k)f_k' \\
&+\left(\frac{x-x_k}{x_{k+1}-x_k}\right)^2(x-x_{k+1})f_{k+1}' .
\end{aligned}
\tag{6.5}
$$

若在整个区间$[a,b]$上定义一组分段三次插值基函数$\alpha_j(x)$及$\beta_j(x)(j=0,1,\cdots,n)$，则$I_h(x)$可表示为

$$
I_h(x)=\sum_{j=0}^{n}[f_j\alpha_j(x)+f_j'\beta_j(x)], \tag{6.6}
$$

其中$\alpha_j(x)$，$\beta_j(x)$分别表示为

$$
\alpha_j(x)=\begin{cases}
\left(\dfrac{x-x_{j-1}}{x_j-x_{j-1}}\right)^2\left(1+2\dfrac{x-x_j}{x_{j-1}-x_j}\right), & x_{j-1}\leqslant x\leqslant x_j \\ & (j=0\text{ 略去}), \\
\left(\dfrac{x-x_{j+1}}{x_j-x_{j+1}}\right)^2\left(1+2\dfrac{x-x_j}{x_{j+1}-x_j}\right), & x_j\leqslant x\leqslant x_{j+1} \\ & (j=n\text{ 略去}), \\
0, & \text{其他};
\end{cases}
\tag{6.7}
$$

$$
\beta_j(x)=\begin{cases}
\left(\dfrac{x-x_{j-1}}{x_j-x_{j-1}}\right)^2(x-x_j), & x_{j-1}\leqslant x\leqslant x_j(j=0\text{ 略去}), \\
\left(\dfrac{x-x_{j+1}}{x_j-x_{j+1}}\right)^2(x-x_j), & x_j\leqslant x\leqslant x_{j+1}(j=n\text{ 略去}), \\
0, & \text{其他}.
\end{cases}
\tag{6.8}
$$

由于 $\alpha_j(x)$，$\beta_j(x)$ 的局部非零性质，当 $x\in[x_k,x_{k+1}]$ 时，只有 $\alpha_k(x)$，$\alpha_{k+1}(x)$，$\beta_k(x)$，$\beta_{k+1}(x)$ 不为零，于是(6.6)的 $I_h(x)$ 可表示为

$$I_h(x) = f_k\alpha_k(x) + f_{k+1}\alpha_{k+1}(x) + f_k'\beta_k(x) + f_{k+1}'\beta_{k+1}(x) \quad (x_k \leqslant x \leqslant x_{k+1}). \tag{6.9}$$

为了研究 $I_h(x)$ 的收敛性，由(6.7)及(6.8)直接得估计式

$$0 \leqslant \alpha_j(x) \leqslant 1, \tag{6.10}$$

$$|\beta_k(x)| \leqslant \frac{4}{27}h_k, \quad |\beta_{k+1}(x)| \leqslant \frac{4}{27}h_k. \tag{6.11}$$

此外，当 $f(x)$ 是分段三次多项式时，$f(x)$ 的插值多项式 $I_h(x)$ 就是它本身. 例如，当 $f(x)=1$ 时就有

$$\sum_{j=0}^{n}\alpha_j(x) = 1.$$

当 $x\in[x_k,x_{k+1}]$ 时，就得

$$\alpha_k(x) + \alpha_{k+1}(x) = 1. \tag{6.12}$$

由(6.9)～(6.12)，当 $x\in[x_k,x_{k+1}]$ 时还可得

$$\begin{aligned}|f(x) - I_h(x)| &\leqslant \alpha_k(x)|f(x) - f_k| + \alpha_{k+1}(x)|f(x) - f_{k+1}| \\ &\quad + (4/27)h_k[|f_k'| + |f_{k+1}'|] \\ &\leqslant \alpha_k(x)|f'(\xi)|h_k + \alpha_{k+1}(x)|f'(\eta)|h_{k+1} \\ &\quad + \frac{4}{27}[|f_k'| + |f_{k+1}'|]h_k,\end{aligned}$$

这里 $\xi,\eta\in(x_k,x_{k+1})$ 且依赖于 x. 因此对 $\forall\, x\in[a,b]$ 成立

$$\max_{a\leqslant x\leqslant b}|f(x) - I_h(x)| \leqslant \frac{35}{27}h\max_{a\leqslant x\leqslant b}|f'(x)|. \tag{6.13}$$

这表明用 $I_h(x)$ 逼近 $f(x)$ 时，它的界只依赖 h，而与 x 无关. 因此，当 $x\in[a,b]$ 时

$$\lim_{h\to 0}I_h(x) = f(x)$$

一致成立. 从而得到：

定理 3 设 $f\in C^1[a,b]$，则当 $h\to 0$ 时，$I_h(x)$ 在 $[a,b]$ 上一致

收敛于 $f(x)$.

2.7 三次样条插值

上面讨论的分段低次插值函数都有一致收敛性，但光滑性较差，对于像高速飞机的机翼形线，船体放样等型值线往往要求有二阶光滑度，即有二阶连续导数. 早期工程师制图时，把富有弹性的细长木条（所谓样条）用压铁固定在样点上，在其他地方让它自由弯曲，然后画下长条的曲线，称为样条曲线. 样条曲线实际上是由分段三次曲线并接而成，在连接点即样点上要求二阶导数连续，从数学上加以概括就得到数学样条这一概念. 下面我们讨论最常用的三次样条函数.

2.7.1 三次样条函数

定义 4 若函数 $S(x)\in C^2[a,b]$，且在每个小区间 $[x_j,x_{j+1}]$ 上是三次多项式，其中 $a=x_0<x_1<\cdots<x_n=b$ 是给定节点，则称 $S(x)$ 是节点 $x_0,x_1,\cdots,x_n$ 上的**三次样条函数**. 若在节点 x_j 上给定函数值 $y_j=f(x_j)(j=0,1,\cdots,n)$，并成立

$$S(x_j)=y_j \qquad (j=0,1,\cdots,n), \tag{7.1}$$

则称 $S(x)$ 为**三次样条插值函数**.

从定义知要求出 $S(x)$，在每个小区间 $[x_j,x_{j+1}]$ 上要确定 4 个待定系数，共有 n 个小区间，故应确定 $4n$ 个参数. 根据 $S(x)$ 在 $[a,b]$ 上二阶导数连续，在节点 $x_j(j=1,2,\cdots,n-1)$ 处应满足连续性条件

$$\begin{aligned}&S(x_j-0)=S(x_j+0),\ S'(x_j-0)=S'(x_j+0),\\&S''(x_j-0)=S''(x_j+0).\end{aligned} \tag{7.2}$$

共有 $3n-3$ 个条件，再加上 $S(x)$ 满足插值条件 (7.1)，共有 $4n-2$ 个条件，因此还需要 2 个条件才能确定 $S(x)$. 通常可在区间 $[a,b]$

端点 $a=x_0, b=x_n$ 上各加一个条件(称为**边界条件**),可根据实际问题的要求给定. 常见的有以下 3 种:

1° 已知两端的一阶导数值,即

$$S'(x_0) = f_0', \ S'(x_n) = f_n'. \tag{7.3}$$

2° 两端的二阶导数已知,即

$$S''(x_0) = f_0'', \ S''(x_n) = f_n'', \tag{7.4}$$

其特殊情况为

$$S''(x_0) = S''(x_n) = 0. \tag{7.4'}$$

(7.4)′称为**自然边界条件**.

3° 当 $f(x)$是以 $x_n - x_0$ 为周期的周期函数时,则要求 $S(x)$也是周期函数. 这时边界条件应满足

$$S(x_0+0) = S(x_n-0), \ S'(x_0+0) = S'(x_n-0),$$
$$S''(x_0+0) = S''(x_n-0), \tag{7.5}$$

而此时(7.1)中 $y_0=y_n$. 这样确定的样条函数 $S(x)$称为**周期样条函数**.

2.7.2 样条插值函数的建立

构造满足插值条件(7.1)及相应边界条件的三次样条插值函数 $S(x)$的表达式可以有多种方法. 例如,可以直接利用分段三次埃尔米特插值(6.6),只要假定 $S'(x_j)=m_j(j=0,1,\cdots,n)$,再由(7.1)可得

$$S(x) = \sum_{j=0}^{n} [y_j\alpha_j(x) + m_j\beta_j(x)], \tag{7.6}$$

其中 $\alpha_j(x), \beta_j(x)$是由(6.7),(6.8)表示的插值基函数,利用条件(7.2)及相应边界条件(7.3)~(7.5)则可得到关于 $m_j(j=0,1,\cdots,n)$的三对角方程组,求出 m_j 则得到所求的三次样条函数 $S(x)$.

下面我们利用 $S(x)$的二阶导数值 $S''(x_j)=M_j(j=0,1,\cdots,$

n)表达 $S(x)$，由于 $S(x)$ 在区间$[x_j, x_{j+1}]$上是三次多项式，故 $S''(x)$在$[x_j, x_{j+1}]$上是线性函数，可表示为

$$S''(x) = M_j \frac{x_{j+1} - x}{h_j} + M_{j+1} \frac{x - x_j}{h_j}. \qquad (7.7)$$

对 $S''(x)$积分两次并利用 $S(x_j) = y_j$ 及 $S(x_{j+1}) = y_{j+1}$，可定出积分常数，于是得三次样条表达式

$$\begin{aligned} S(x) &= M_j \frac{(x_{j+1} - x)^3}{6h_j} + M_{j+1} \frac{(x - x_j)^3}{6h_j} \\ &+ \left(y_j - \frac{M_j h_j^2}{6}\right) \frac{x_{j+1} - x}{h_j} + \left(y_{j+1} - \frac{M_{j+1} h_j^2}{6}\right) \frac{x - x_j}{h_j} \\ &(j = 0, 1, \cdots, n-1). \end{aligned} \qquad (7.8)$$

这里 $M_j, j = 0, 1, \cdots, n$，是未知的. 为了确定 $M_j, j = 0, 1, \cdots, n$，对 $S(x)$求导得

$$\begin{aligned} S'(x) &= -M_j \frac{(x_{j+1} - x)^2}{2h_j} + M_{j+1} \frac{(x - x_j)^2}{2h_j} \\ &+ \frac{y_{j+1} - y_j}{h_j} - \frac{M_{j+1} - M_j}{6} h_j; \end{aligned} \qquad (7.9)$$

由此可求得

$$S'(x_j + 0) = -\frac{h_j}{3} M_j - \frac{h_j}{6} M_{j+1} + \frac{y_{j+1} - y_j}{h_j}.$$

类似地可求出 $S(x)$在区间$[x_{j-1}, x_j]$上的表达式，从而得

$$S'(x_j - 0) = \frac{h_{j-1}}{6} M_{j-1} + \frac{h_{j-1}}{3} M_j + \frac{y_j - y_{j-1}}{h_{j-1}},$$

利用 $S'(x_j + 0) = S'(x_j - 0)$可得

$$\mu_j M_{j-1} + 2M_j + \lambda_j M_{j+1} = d_j \quad (j = 1, 2, \cdots, n-1), \qquad (7.10)$$

其中

$$\mu_j = \frac{h_{j-1}}{h_{j-1} + h_j}, \ \lambda_j = \frac{h_j}{h_{j-1} + h_j}, \ j = 0, 1, \cdots, n,$$

$$d_j = 6\frac{f[x_j, x_{j+1}] - f[x_{j-1}, x_j]}{h_{j-1} + h_j} = 6f[x_{j-1}, x_j; x_{j+1}]. \tag{7.11}$$

对第一种边界条件(7.3),可导出两个方程

$$\left.\begin{aligned} 2M_0 + M_1 &= \frac{6}{h_0}(f[x_0, x_1] - f_0'), \\ M_{n-1} + 2M_n &= \frac{6}{h_{n-1}}(f_n' - f[x_{n-1}, x_n]). \end{aligned}\right\} \tag{7.12}$$

如果令 $\lambda_0 = 1, d_0 = \frac{6}{h_0}(f[x_0, x_1] - f_0'), \mu_n = 1, d_n = \frac{6}{h_{n-1}}(f_n' - f[x_{n-1}, x_n])$,那么(7.10)及(7.12)可写成矩阵形式

$$\begin{pmatrix} 2 & \lambda_0 & & & \\ \mu_1 & 2 & \lambda_1 & & \\ & \ddots & \ddots & \ddots & \\ & & \mu_{n-1} & 2 & \lambda_{n-1} \\ & & & \mu_n & 2 \end{pmatrix} \begin{pmatrix} M_0 \\ M_1 \\ \vdots \\ M_{n-1} \\ M_n \end{pmatrix} = \begin{pmatrix} d_0 \\ d_1 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}. \tag{7.13}$$

对第二种边界条件(7.4),直接得端点方程

$$M_0 = f_0'', \qquad M_n = f_n''. \tag{7.14}$$

如果令 $\lambda_0 = \mu_n = 0, d_0 = 2f_0'', d_n = 2f_n''$,则(7.10)和(7.14)也可以写成(7.13)的形式.

对于第三种边界条件(7.5),可得

$$M_0 = M_n, \qquad \lambda_n M_1 + \mu_n M_{n-1} + 2M_n = d_n, \tag{7.15}$$

其中 $\lambda_n = \frac{h_0}{h_{n-1} + h_0}$, $\mu_n = 1 - \lambda_n = \frac{h_{n-1}}{h_{n-1} + h_0}$, $d_n = 6\frac{f[x_0, x_1] - f[x_{n-1}, x_n]}{h_0 + h_{n-1}}$,(7.10)和(7.15)可以写成矩阵形式

$$\begin{pmatrix} 2 & \lambda_1 & & & \mu_1 \\ \mu_2 & 2 & \lambda_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \mu_{n-1} & 2 & \lambda_{n-1} \\ \lambda_n & & & \mu_n & 2 \end{pmatrix} \begin{pmatrix} M_1 \\ M_2 \\ \vdots \\ M_{n-1} \\ M_n \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}. \tag{7.16}$$

(7.13)和(7.16)是关于 $M_j(j=0,1,\cdots,n)$的三对角方程组，M_j 在力学上解释为细梁在 x_j 截面处的弯矩，称为 $S(x)$的矩，方程组(7.13)和(7.16)称为三弯矩方程. (7.13)和(7.16)的系数矩阵中元素 λ_j,μ_j 已完全确定. 并且满足 $\lambda_j\geqslant 0,\mu_j\geqslant 0,\lambda_j+\mu_j=1$. 因此系数矩阵为严格对角占优阵，从而(7.13)和(7.16)有唯一解. 求解方法可见第 5 章第 4 节追赶法，将解得结果代入(7.8)的表达式即可.

例 5 设 $f(x)$为定义在[27.7,30]上的函数，在节点 $x_i(i=0,1,2,3)$上的值如下：

$f(x_0)=f(27.7)=4.1, f(x_1)=f(28)=4.3,$

$f(x_2)=f(29)=4.1, f(x_3)=f(30)=3.0.$

试求三次样条函数 $S(x)$，使它满足边界条件 $S'(27.7)=3.0$，$S'(30)=-4.0$.

解 先由(7.11)及(7.12)计算 $h_0=0.30, h_1=h_2=1, \mu_1=\frac{3}{13}, \mu_2=\frac{1}{2}, \mu_3=1, \lambda_0=1, \lambda_1=\frac{10}{13}, \lambda_2=\frac{1}{2}, d_0=\frac{6}{h_0}(f[x_0,x_1]-f_0')$ $=-46.666, d_1=6f[x_0,x_1,x_2]=-4.00002, d_2=6f[x_1,x_2,x_3]$ $=-2.70000, d_3=\frac{6}{h_2}(f_3'-f[x_2,x_3])=-17.4$.

由此得矩阵形式的方程组(7.13)为

$$\begin{pmatrix} 2 & 1 & & \\ \frac{3}{13} & 2 & \frac{10}{13} & \\ & \frac{1}{2} & 2 & \frac{1}{2} \\ & & 1 & 2 \end{pmatrix}\begin{pmatrix} M_0 \\ M_1 \\ M_2 \\ M_3 \end{pmatrix}=\begin{pmatrix} -46.6666 \\ -4.00002 \\ -2.7000 \\ -17.4000 \end{pmatrix}.$$

求解此方程组得到

$$M_0=-23.531,\ M_1=0.395,\ M_2=0.830,\ M_3=-9.115.$$

将 M_0,M_1,M_2,M_3 代入表达式(7.8)得到(曲线见图 2-6)

$$S(x)=\begin{cases}13.07278(x-28)^3-14.84322(x-28)+0.21944 \\ \quad \cdot (x-27.7)^3+14.31358(x-27.7), \quad x\in[27.7,28], \\ 0.06583(29-x)^3+4.23417(29-x)+0.13833 \\ \quad \cdot (x-28)^3+3.96167(x-28), \quad x\in[28,29], \\ 0.13833(30-x)^3+3.96167(30-x)-1.51917 \\ \quad \cdot (x-29)^3+4.51917(x-29), \quad x\in[29,30].\end{cases}$$

通常求三次样条函数可根据上述例题的计算步骤直接编程上机计算，或直接使用数学库中软件，根据具体要求算出结果即可.

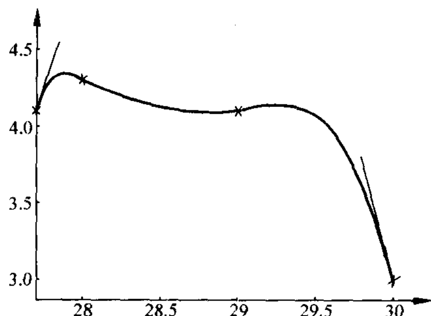


图　2-6

例 6　给定函数 $f(x)=\frac{1}{1+x^2}$，$-5\leqslant x\leqslant 5$，节点 $x_k=-5+k$ $(k=0,1,\cdots,10)$，用三次样条插值求 $S_{10}(x)$.

取 $S_{10}(x_k)=f(x_k)\quad(k=0,1,\cdots,10)$，$S'_{10}(-5)=f'(-5)$，$S'_{10}(5)=f'(5)$. 直接上机计算可求出 $S_{10}(x)$ 在表 2-6 所列各点的值. 从表中看到，在所列各点 $S_{10}(x)$ 与 $f(x)$ 误差较小，它可作为 $f(x)$ 在区间 $[-5,5]$ 上的近似，而用拉格朗日插值多项式 $L_{10}(x)$ 计算相应点上的值 $L_{10}(x)$（也见表 2-6），显然它与 $f(x)$ 相差很大，在图 2-5 中已经看到它不能作为 $f(x)$ 的近似.

表 2-6

x	$\frac{1}{1+x^2}$	$S_{10}(x)$	$L_{10}(x)$	x	$\frac{1}{1+x^2}$	$S_{10}(x)$	$L_{10}(x)$
−5.0	0.03846	0.03846	0.03846	−2.3	0.15898	0.16115	0.24145
−4.8	0.04160	0.03758	1.80438	−2.0	0.20000	0.20000	0.20000
−4.5	0.04706	0.04248	1.57872	−1.8	0.23585	0.23154	0.18878
−4.3	0.05131	0.04842	0.88808	−1.5	0.30769	0.29744	0.23535
−4.0	0.05882	0.05882	0.05882	−1.3	0.37175	0.36133	0.31650
−3.8	0.06477	0.06556	−0.20130	−1.0	0.50000	0.50000	0.50000
−3.5	0.07547	0.07606	−0.22620	−0.8	0.60976	0.62420	0.64316
−3.3	0.08410	0.08426	−0.10832	−0.5	0.80000	0.82051	0.84340
−3.0	0.10000	0.10000	0.10000	−0.3	0.91743	0.92754	0.94090
−2.8	0.11312	0.11366	0.19837	0	1.00000	1.00000	1.00000
−2.5	0.13793	0.13971	0.25376				

2.7.3 误差界与收敛性

三次样条函数的收敛性与误差估计比较复杂，这里不加证明地给出一个主要结果.

定理 4 设 $f(x)\in C^4[a,b]$，$S(x)$为满足第一种或第二种边界条件(7.3)或(7.4)的三次样条函数，令 $h=\max\limits_{0\leqslant i\leqslant n-1} h_i$，$h_i=x_{i+1}-x_i\ (i=0,1,\cdots,n-1)$，则有估计式

$$\max_{a\leqslant x\leqslant b}|f^{(k)}(x)-S^{(k)}(x)|\leqslant C_k\max_{a\leqslant x\leqslant b}|f^{(4)}(x)|h^{4-k},\ k=0,1,2,\tag{7.17}$$

其中 $C_0=\frac{5}{384}$，$C_1=\frac{1}{24}$，$C_2=\frac{3}{8}$.

这个定理不但给出了三次样条插值函数 $S(x)$的误差估计，且当 $h\to 0$ 时，$S(x)$及其一阶导数 $S'(x)$和二阶导数 $S''(x)$均分别一

致收敛于 $f(x)$，$f'(x)$及 $f''(x)$.

评 注

插值法是一个古老而实用的课题. 它是函数逼近，数值微积分和微分方程数值解的基础. 本章讨论了拉格朗日插值公式及牛顿插值公式，前者在理论上较为重要，后者在计算插值多次式及求函数近似值较为方便且节省计算量. 等距节点插值是应用中最常见的，利用差分及牛顿前插与后插公式即可，还有利用中心差分得到的其他类型插值公式，因使用较少本章没有介绍，如有需要可参看文献[3].

对充分光滑的被插函数可采用微分形式的误差估计给出误差限，其他情形可利用均差形式给出误差估计的近似值. 但由于高次插值存在病态性质，一般实际计算很少使用高次插值，更多使用分段低次插值，特别是三次样条插值，由于它具有良好的收敛性和稳定性，又有二阶光滑度，因此在理论上和应用中均有重要意义，本章只对最常用的三弯矩方程做简单介绍，一般的样条函数理论和 B-样条等更详细内容读者如有需要可参看文献[4,5,10]等.

习 题

1. 当 $x=1,-1,2$ 时，$f(x)=0,-3,4$，求 $f(x)$的二次插值多项式.

2. 给出 $f(x)=\ln x$ 的数值表

x	0.4	0.5	0.6
$\ln x$	−0.916291	−0.693147	−0.510826

x	0.7	0.8
$\ln x$	−0.356675	−0.223144

用线性插值及二次插值计算 ln0.54 的近似值.

3. 给出 $\cos x, 0°\leqslant x\leqslant 90°$ 的函数表，步长 $h=1'=(1/60)°$，若函数表具有 5 位有效数字，研究用线性插值求 $\cos x$ 近似值时的总误差界.

4. 设 x_j 为互异节点 $(j=0,1,\cdots,n)$，求证：

i) $\sum_{j=0}^{n} x_j^k l_j(x) \equiv x^k \quad (k=0,1,\cdots,n)$;

ii) $\sum_{j=0}^{n} (x_j - x)^k l_j(x) \equiv 0 \quad (k=1,2,\cdots,n)$.

5. 设 $f(x)\in C^2[a,b]$ 且 $f(a)=f(b)=0$，求证：

$$\max_{a\leqslant x\leqslant b} |f(x)| \leqslant \frac{1}{8}(b-a)^2 \max_{a\leqslant x\leqslant b} |f''(x)|.$$

6. 在 $-4\leqslant x\leqslant 4$ 上给出 $f(x)=e^x$ 的等距节点函数表，若用二次插值求 e^x 的近似值，要使截断误差不超过 10^{-6}，问使用函数表的步长 h 应取多少？

7. 若 $y_n=2^n$，求 $\Delta^4 y_n$ 及 $\delta^4 y_n$.

8. 如果 $f(x)$ 是 m 次多项式，记 $\Delta f(x)=f(x+h)-f(x)$，证明 $f(x)$ 的 k 阶差分 $\Delta^k f(x)(0\leqslant k\leqslant m)$ 是 $m-k$ 次多项式，并且 $\Delta^{m+l} f(x)=0$（l 为正整数）.

9. 证明 $\Delta(f_k g_k)=f_k \Delta g_k + g_{k+1}\Delta f_k$.

10. $\sum_{k=0}^{n-1} f_k \Delta g_k = f_n g_n - f_0 g_0 - \sum_{k=0}^{n-1} g_{k+1}\Delta f_k$.

11. 证明 $\sum_{j=0}^{n-1} \Delta^2 y_j = \Delta y_n - \Delta y_0$.

12. 若 $f(x)=a_0+a_1 x+\cdots+a_{n-1}x^{n-1}+a_n x^n$ 有 n 个不同实根 $x_1, x_2,\cdots,x_n$，证明：

$$\sum_{j=1}^{n} \frac{x_j^k}{f'(x_j)} = \begin{cases} 0, 0\leqslant k\leqslant n-2; \\ a_n^{-1},\ k=n-1. \end{cases}$$

13. 证明 n 阶均差有下列性质：

i) 若 $F(x)=cf(x)$，则 $F[x_0,x_1,\cdots,x_n]=cf[x_0,x_1,\cdots,x_n]$;

ii) 若 $F(x)=f(x)+g(x)$，则

$$F[x_0,x_1,\cdots,x_n]=f[x_0,x_1,\cdots,x_n]+g[x_0,x_1,\cdots,x_n].$$

14. $f(x)=x^7+x^4+3x+1$，求

$$f[2^0,2^1,\cdots,2^7] \text{ 及 } f[2^0,2^1,\cdots,2^8].$$

15. 证明两点三次埃尔米特插值余项是

$$R_3(x)=f^{(4)}(\xi)(x-x_k)^2(x-x_{k+1})^2/4!,\ \xi\in(x_k,x_{k+1}),$$

并由此求出分段三次埃尔米特插值的误差限.

16. 求一个次数不高于 4 次的多项式 $P(x)$,使它满足 $P(0)=P'(0)=0,P(1)=P'(1)=1,P(2)=1$.

17. 设 $f(x)=1/(1+x^2)$,在 $-5\leqslant x\leqslant 5$ 上取 $n=10$,按等距节点求分段线性插值函数 $I_h(x)$,计算各节点间中点处的 $I_h(x)$ 与 $f(x)$ 的值,并估计误差.

18. 求 $f(x)=x^2$ 在 $[a,b]$ 上的分段线性插值函数 $I_h(x)$,并估计误差.

19. 求 $f(x)=x^4$ 在 $[a,b]$ 上的分段埃尔米特插值,并估计误差.

20. 给定数据表如下:

x_j	0.25	0.30	0.39	0.45	0.53
y_j	0.5000	0.5477	0.6245	0.6708	0.7280

试求三次样条插值 $S(x)$,并满足条件:

i) $S'(0.25)=1.0000$, $S'(0.53)=0.6868$;

ii) $S''(0.25)=S''(0.53)=0$.

21. 若 $f(x)\in C^2[a,b]$,$S(x)$是三次样条函数,证明:

i) $$\int_a^b[f''(x)]^2\mathrm{d}x-\int_a^b[S''(x)]^2\mathrm{d}x=\int_a^b[f''(x)-S''(x)]^2\mathrm{d}x+2\int_a^b S''(x)[f''(x)-S''(x)]\mathrm{d}x;$$

ii) 若 $f(x_i)=S(x_i)(i=0,1,\cdots,n)$,式中 x_i 为插值节点,且 $a=x_0<x_1<\cdots<x_n=b$,则

$$\int_a^b S''(x)[f''(x)-S''(x)]\mathrm{d}x=S''(b)[f'(b)-S'(b)]-S''(a)[f'(a)-S'(a)].$$

第3章 函数逼近与曲线拟合

3.1 函数逼近的基本概念

3.1.1 函数逼近与函数空间

在数值计算中经常要计算函数值，如计算机中计算基本初等函数及其他特殊函数；当函数只在有限点集上给定函数值，要在包含该点集的区间上用公式给出函数的简单表达式，这些都涉及到在区间$[a,b]$上用简单函数逼近已知复杂函数的问题，这就是函数逼近问题. 上章讨论的插值法就是函数逼近问题的一种. 本章讨论的函数逼近，是指"对函数类A中给定的函数$f(x)$，记作$f(x)\in A$，要求在另一类简单的便于计算的函数类B中求函数$p(x)\in B$，使$p(x)$与$f(x)$的误差在某种度量意义下最小". 函数类A通常是区间$[a,b]$上的连续函数，记作$C[a,b]$，称为连续函数空间，而函数类B通常为n次多项式，有理函数或分段低次多项式等. 函数逼近是数值分析的基础，为了在数学上描述更精确，先要介绍代数和分析中一些基本概念及预备知识.

数学上常把在各种集合中引入某些不同的确定关系称为赋予集合以某种空间结构，并将这样的集合称为空间. 例如将所有实n维向量组成的集合，按向量加法及向量与数的乘法构成实数域上的线性空间，记作$\mathbf{R}^n$，称为n维向量空间. 类似地，对次数不超过n(n为正整数)的实系数多项式全体，按通常多项式与多项式加法及数与多项式乘法也构成数域$\mathbf{R}$上一个线性空间，用H_n表示，称为多项式空间. 所有定义在$[a,b]$上的连续函数集合，按函数加法

和数与函数乘法构成数域 **R** 上的线性空间，记作 $C[a,b]$. 类似地记 $C^p[a,b]$为具有 p 阶连续导数的函数空间.

定义 1 设集合 S 是数域 **P** 上的线性空间，元素 $x_1,\cdots,x_n\in S$，如果存在不全为零的数 $\alpha_1,\cdots,\alpha_n\in\mathbf{P}$，使得

$$\alpha_1 x_1+\cdots+\alpha_n x_n=0, \tag{1.1}$$

则称 $x_1,\cdots,x_n$ 线性相关. 否则，若等式(1.1)只对 $\alpha_1=\alpha_2=\cdots=\alpha_n=0$ 成立，则称 $x_1,\cdots,x_n$ 线性无关.

若线性空间 S 是由 n 个线性无关元素 $x_1,\cdots,x_n$ 生成的，即对 $\forall\, x\in S$ 都有

$$x=\alpha_1 x_1+\cdots+\alpha_n x_n,$$

则 $x_1,\cdots,x_n$ 称为空间 S 的一组基，记为 $S=\mathrm{span}\{x_1,\cdots,x_n\}$，并称空间 S 为 n 维空间，系数 $\alpha_1,\cdots,\alpha_n$ 称为 x 在基 $x_1,\cdots,x_n$ 下的坐标，记作$(\alpha_1,\cdots,\alpha_n)$，如果 S 中有无限个线性无关元素 $x_1,\cdots,x_n,\cdots$，则称 S 为无限维线性空间.

下面考察次数不超过 n 次的多项式集合 H_n，其元素 $p(x)\in H_n$ 表示为

$$p(x)=a_0+a_1x+\cdots+a_nx^n, \tag{1.2}$$

它由 $n+1$ 个系数$(a_0,a_1,\cdots,a_n)$唯一确定. $1,x,\cdots,x^n$ 线性无关，它是 H_n 的一组基，故 $H_n=\mathrm{span}\{1,x,\cdots,x^n\}$，且$(a_0,a_1,\cdots,a_n)$是$p(x)$的坐标向量，$H_n$ 是 $n+1$ 维的.

对连续函数 $f(x)\in C[a,b]$，它不能用有限个线性无关的函数表示，故 $C[a,b]$是无限维的，但它的任一元素 $f(x)\in C[a,b]$均可用有限维的 $p(x)\in H_n$ 逼近，使误差$\max\limits_{a\leqslant x\leqslant b}|f(x)-p(x)|<\varepsilon$（$\varepsilon$ 为任给的小正数），这就是著名的魏尔斯特拉斯(Weierstrass)定理.

定理 1 设 $f(x)\in C[a,b]$，则对任何 $\varepsilon>0$，总存在一个代数多项式 $p(x)$，使

$$\|f(x)-p(x)\|_\infty<\varepsilon$$

在$[a,b]$上一致成立.

这定理已在“数学分析”中证明过. 这里需要说明的是在许多证明方法中，伯恩斯坦(Бернштейн)1912 年给出的证明是一种构造性证明. 他根据函数整体逼近的特性构造出伯恩斯坦多项式

$$B_n(f,x) = \sum_{k=0}^{n} f\left(\frac{k}{n}\right) P_k(x); \tag{1.3}$$

其中

$$P_k(x) = \binom{n}{k} x^k (1-x)^{n-k},$$

$\binom{n}{k} = \frac{n(n-1)\cdots(n-k+1)}{k!}$ 为二项式展开系数，并证明了 $\lim\limits_{n\to\infty} B_n(f,x) = f(x)$ 在 $[0,1]$ 上一致成立；若 $f(x)$ 在 $[0,1]$ 上 m 阶导数连续，则

$$\lim_{n\to\infty} B_n^{(m)}(f,x) = f^{(m)}(x).$$

这不但证明了定理 1，而且由(1.3)给出了 $f(x)$ 的一个逼近多项式. 它与拉格朗日插值多项式

$$L_n(x) = \sum_{k=0}^{n} f(x_k) l_k(x), \sum_{k=0}^{n} l_k(x) = 1$$

很相似，对 $B_n(f,x)$，当 $f(x)=1$ 时也有关系式

$$\sum_{k=0}^{n} P_k(x) = \sum_{k=0}^{n} \binom{n}{k} x^k (1-x)^{n-k} = 1. \tag{1.4}$$

这只要在恒等式

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}$$

中令 $y=1-x$ 就可得到. 但这里当 $x\in[0,1]$ 时还有 $P_k(x)\geqslant 0$，于是

$$\sum_{k=0}^{n} |P_k(x)| = \sum_{k=0}^{n} P_k(x) = 1$$

是有界的，因而只要 $|f(x)|\leqslant\delta$ 对任意 $x\in[0,1]$ 成立，则

$$|B_n(f,x)| \leqslant \max_{0\leqslant x\leqslant 1} |f(x)| \sum_{k=0}^{n} |P_k(x)| \leqslant \delta$$

有界,故 $B_n(f,x)$是稳定的. 至于拉格朗日插值多项式 $L_n(x)$,由于 $\sum_{k=0}^{n} |l_k(x)|$ 无界,因而不能保证高阶插值的稳定性与收敛性. 相比之下,多项式 $B_n(f,x)$有良好的逼近性质,但它收敛太慢,比三次样条逼近效果差得多,实际中很少被使用.

更一般地,可用一组在 $C[a,b]$上线性无关的函数集合 $\{\varphi_i(x)\}_{i=0}^{n}$来逼近 $f(x)\in C[a,b]$,元素 $\varphi(x)\in \Phi=\mathrm{span}\{\varphi_0(x), \varphi_1(x),\cdots,\varphi_n(x)\}\subset C[a,b]$,表示为

$$\varphi(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + \cdots + a_n\varphi_n(x). \qquad (1.5)$$

函数逼近问题就是对任何 $f\in C[a,b]$,在子空间 Φ 中找一个元素 $\varphi^*(x)\in \Phi$,使 $f(x)-\varphi^*(x)$在某种意义下最小.

3.1.2　范数与赋范线性空间

为了对线性空间中元素大小进行衡量,需要引进范数定义,它是 $\mathbf{R}^n$ 空间中向量长度概念的直接推广.

定义 2　设 S 为线性空间,$x\in S$,若存在唯一实数 $\|\cdot\|$,满足条件:

(1) $\|x\| \geqslant 0$,当且仅当 $x=0$ 时,$\|x\|=0$;　(正定性)

(2) $\|\alpha x\| = |\alpha|\,\|x\|$,　$\alpha\in\mathbf{R}$;　(齐次性)

(3) $\|x+y\| \leqslant \|x\| + \|y\|$,　$x,y\in S$. (三角不等式)

则称 $\|\cdot\|$ 为线性空间 S 上的范数,S 与 $\|\cdot\|$ 一起称为赋范线性空间,记为 X.

例如,在 $\mathbf{R}^n$ 上的向量 $\boldsymbol{x}=(x_1,\cdots,x_n)^{\mathrm{T}}\in\mathbf{R}^n$,三种常用范数为

$$\|x\|_\infty = \max_{1\leqslant i\leqslant n} |x_i| ,$$　称为∞-范数或最大范数,

$$\|x\|_1 = \sum_{i=1}^{n} |x_i|, \qquad \text{称为 1-范数,}$$

$$\|x\|_2 = \left(\sum_{i=1}^{n} x_i^2\right)^{\frac{1}{2}}, \qquad \text{称为 2-范数.}$$

类似地对连续函数空间 $C[a,b]$,若 $f\in C[a,b]$可定义三种常用范数如下:

$$\|f\|_\infty = \max_{a\leqslant x\leqslant b} |f(x)|, \qquad \text{称为}\infty\text{-范数,}$$

$$\|f\|_1 = \int_a^b |f(x)|\,\mathrm{d}x, \qquad \text{称为 1-范数,}$$

$$\|f\|_2 = \left(\int_a^b f^2(x)\mathrm{d}x\right)^{\frac{1}{2}}, \qquad \text{称为 2-范数.}$$

可以验证这样定义的范数均满足定义 2 中的三个条件.

3.1.3 内积与内积空间

在线性代数中,$\mathbf{R}^n$ 中两个向量 $\boldsymbol{x}=(x_1,\cdots,x_n)^{\mathrm{T}}$ 及 $\boldsymbol{y}=(y_1,\cdots,y_n)^{\mathrm{T}}$的内积定义为

$$(x,y) = x_1y_1 + \cdots + x_ny_n.$$

若将它推广到一般的线性空间 X,则有下面的定义.

定义 3 设 X 是数域 $\mathbf{K}$($\mathbf{R}$ 或 $\mathbf{C}$)上的线性空间,对 $\forall\, u,v\in X$,有 $\mathbf{K}$ 中一个数与之对应,记为(u,v),它满足以下条件:

(1) $(u,v)=\overline{(v,u)}$, $\qquad \forall\, u,v\in X$;

(2) $(\alpha u,v)=\alpha(u,v)$, $\quad \alpha\in \mathbf{K}, u,v\in X$;

(3) $(u+v,w)=(u,w)+(v,w)$, $\quad \forall\, u,v,w\in X$;

(4) $(u,u)\geqslant 0$,当且仅当 $u=0$ 时,$(u,u)=0$.

则称(u,v)为 X 上 u 与 v 的内积. 定义了内积的线性空间称为内积空间. 定义中(1)的右端$\overline{(u,v)}$称为(u,v)的共轭,当 $\mathbf{K}$ 为实数域 $\mathbf{R}$ 时$(u,v)=(v,u)$.

如果$(u,v)=0$,则称 u 与 v 正交,这是向量相互垂直概念的推

广. 关于内积空间性质有以下重要定理.

定理 2 设 X 为一个内积空间,对 $\forall u,v\in X$,有

$$|(u,v)|^2 \leqslant (u,u)(v,v). \tag{1.6}$$

称为柯西-施瓦茨(Cauchy-Schwarz)不等式.

证明 当 $v=0$ 时(1.6)式显然成立. 现设 $v\neq 0$,则 $(v,v)>0$,且对任何数 λ 有

$$0\leqslant (u+\lambda v,u+\lambda v)=(u,u)+2\lambda(u,v)+\lambda^2(v,v).$$

取 $\lambda=-(u,v)/(v,v)$,代入上式右端,得

$$(u,u)-2\frac{|(u,v)|^2}{(v,v)}+\frac{|(u,v)|^2}{(v,v)}\geqslant 0,$$

即得 $v\neq 0$ 时

$$|(u,v)|^2\leqslant (u,u)(v,v).$$

证毕.

定理 3 设 X 为一个内积空间,$u_1,u_2,\cdots,u_n\in X$,矩阵

$$\boldsymbol{G}=\begin{bmatrix}(u_1,u_1) & (u_2,u_1) & \cdots & (u_n,u_1)\\ (u_1,u_2) & (u_2,u_2) & \cdots & (u_n,u_2)\\ \vdots & \vdots & & \vdots\\ (u_1,u_n) & (u_2,u_n) & \cdots & (u_n,u_n)\end{bmatrix} \tag{1.7}$$

称为格拉姆(Gram)矩阵,则 $\boldsymbol{G}$ 非奇异的充分必要条件是 u_1,$u_2,\cdots,u_n$ 线性无关.

证明 $\boldsymbol{G}$ 非奇异等价于 $\det\boldsymbol{G}\neq 0$,其充要条件是齐次方程组

$$\left(\sum_{j=1}^{n}\alpha_j u_j,u_k\right)=\sum_{j=1}^{n}(u_j,u_k)\alpha_j=0,\quad k=1,2,\cdots,n \tag{1.8}$$

只有零解;而

$$\sum_{j=1}^{n}\alpha_j u_j=\alpha_1u_1+\alpha_2u_2+\cdots+\alpha_nu_n=0 \tag{1.9}$$

$$\Leftrightarrow\left(\sum_{j=1}^{n}\alpha_j u_j,\sum_{j=1}^{n}\alpha_j u_j\right)=0$$

$$\Leftrightarrow\left(\sum_{j=1}^{n}\alpha_j u_j,u_k\right)=0,\quad k=1,2,\cdots,n.$$

从以上等价关系可知，$\det\mathbf{G}\neq 0$ 等价于从(1.8)推出 $\alpha_1=\alpha_2=\cdots=\alpha_n=0$，而后者等价于从(1.9)推出 $\alpha_1=\alpha_2=\cdots=\alpha_n=0$，即 $u_1,u_2,\cdots,u_n$ 线性无关. 证毕.

在内积空间 X 上可以由内积导出一种范数，即对于 $u\in X$，记

$$\|u\|=\sqrt{(u,u)},\tag{1.10}$$

容易验证它满足范数定义的三条性质，其中三角不等式

$$\|u+v\|\leqslant\|u\|+\|v\|\tag{1.11}$$

可由定理2直接得出，即

$$\begin{aligned}(\|u\|+\|v\|)^2&=\|u\|^2+2\|u\|\|v\|+\|v\|^2\\&\geqslant(u,u)+2(u,v)+(v,v)\\&=(u+v,u+v)=\|u+v\|^2,\end{aligned}$$

两端开方即得(1.11).

例1 $\mathbf{R}^n$ 与 $\mathbf{C}^n$ 的内积. 设 $x,y\in\mathbf{R}^n$，$\boldsymbol{x}=(x_1,\cdots,x_n)^{\mathrm{T}}$，$\boldsymbol{y}=(y_1,\cdots,y_n)^{\mathrm{T}}$，则其内积定义为

$$(x,y)=\sum_{i=1}^{n}x_iy_i.\tag{1.12}$$

由此导出的向量2-范数为

$$\|x\|_2=(x,x)^{\frac{1}{2}}=\Big(\sum_{i=1}^{n}x_i^2\Big)^{\frac{1}{2}}.$$

若给定实数 $\omega_i>0(i=1,\cdots,n)$，称 $\{\omega_i\}$ 为权系数，则在 $\mathbf{R}^n$ 上可定义加权内积为

$$(x,y)=\sum_{i=1}^{n}\omega_ix_iy_i,\tag{1.13}$$

相应的范数为

$$\|x\|_2=\Big(\sum_{i=1}^{n}\omega_ix_i^2\Big)^{\frac{1}{2}}.$$

不难验证(1.13)给出的 (x,y) 满足内积定义的4条性质. 当 $\omega_i=1(i=1,\cdots,n)$ 时，(1.13)就是(1.12).

如果 $x,y\in \mathbf{C}^n$，带权内积定义为

$$(x,y)=\sum_{i=1}^{n}\omega_i x_i \bar{y}_i, \tag{1.14}$$

这里$\{\omega_i\}$仍为正实数序列，$\bar{y}_i$ 为 y_i 的共轭.

在 $C[a,b]$上也可以类似定义带权内积，为此先给出权函数的定义.

定义 4 设$[a,b]$是有限或无限区间，在$[a,b]$上的非负函数$\rho(x)$满足条件：

(1) $\int_a^b x^k\rho(x)\mathrm{d}x$ 存在且为有限值($k=0,1,\cdots$)，

(2) 对$[a,b]$上的非负连续函数 $g(x)$，如果 $\int_a^b g(x)\rho(x)\mathrm{d}x=0$，则 $g(x)\equiv 0$.

则称 $\rho(x)$为$[a,b]$上的一个权函数.

例 2 $C[a,b]$上的内积. 设 $f(x),g(x)\in C[a,b]$，$\rho(x)$是$[a,b]$上给定的权函数，则可定义内积

$$(f(x),g(x))=\int_a^b \rho(x)f(x)g(x)\mathrm{d}x. \tag{1.15}$$

容易验证它满足内积定义的 4 条性质，由此内积导出的范数为

$$\| f(x)\|_2=(f(x),f(x))^{\frac{1}{2}}=\left[\int_a^b\rho(x)f^2(x)\mathrm{d}x\right]^{\frac{1}{2}} \tag{1.16}$$

称(1.15)和(1.16)为带权 $\rho(x)$的内积和范数，特别常用的是 $\rho(x)\equiv 1$ 的情形，即

$$(f(x),g(x))=\int_a^b f(x)g(x)\mathrm{d}x,$$

$$\| f(x)\|_2=\left(\int_a^b f^2(x)\mathrm{d}x\right)^{\frac{1}{2}}.$$

若 $\varphi_0,\varphi_1,\cdots,\varphi_n$ 是 $C[a,b]$中的线性无关函数族，记 $\varphi=\mathrm{span}\{\varphi_0,\varphi_1,\cdots,\varphi_n\}$，它的格拉姆矩阵为

$$G = G(\varphi_0, \varphi_1, \cdots, \varphi_n) = \begin{bmatrix} (\varphi_0, \varphi_0) & (\varphi_0, \varphi_1) & \cdots & (\varphi_0, \varphi_n) \\ (\varphi_1, \varphi_0) & (\varphi_1, \varphi_1) & \cdots & (\varphi_1, \varphi_n) \\ \vdots & \vdots & & \vdots \\ (\varphi_n, \varphi_0) & (\varphi_n, \varphi_1) & \cdots & (\varphi_n, \varphi_n) \end{bmatrix}. \tag{1.17}$$

根据定理 3 可知 $\varphi_0, \varphi_1, \cdots, \varphi_n$ 线性无关的充要条件是 $\det G(\varphi_0, \varphi_1, \cdots, \varphi_n) \neq 0$.

3.2 正交多项式

正交多项式是函数逼近的重要工具,在数值积分中也有重要应用.

3.2.1 正交函数族与正交多项式

定义 5 若 $f(x), g(x) \in C[a,b]$,$\rho(x)$为$[a,b]$上的权函数且满足

$$(f(x), g(x)) = \int_a^b \rho(x) f(x) g(x) \mathrm{d}x = 0, \tag{2.1}$$

则称 $f(x)$ 与 $g(x)$ 在$[a,b]$上带权 $\rho(x)$ **正交**. 若函数族 $\varphi_0(x), \varphi_1(x), \cdots, \varphi_n(x), \cdots$满足关系

$$(\varphi_j, \varphi_k) = \int_a^b \rho(x) \varphi_j(x) \varphi_k(x) \mathrm{d}x = \begin{cases} 0, & j \neq k, \\ A_k > 0, & j = k. \end{cases} \tag{2.2}$$

则称$\{\varphi_k(x)\}$是$[a,b]$上带权 $\rho(x)$的**正交函数族**;若 $A_k \equiv 1$,则称之为**标准正交函数族**.

例如,三角函数族

$$1, \cos x, \sin x, \cos 2x, \sin 2x, \cdots$$

就是在区间$[-\pi, \pi]$上的正交函数族. 因为对 $k=1,2,\cdots$有

$$(1,1) = 2\pi, \qquad (\sin kx, \sin kx) = (\cos kx, \cos kx) = \pi,$$

而对 $k,j=1,2,\cdots$，当 $k\neq j$ 时有 $(\cos kx, \sin kx)=(1,\cos kx)=(1,\sin kx)=0$；

$$(\cos kx, \cos jx) = (\sin kx, \sin jx) = (\cos kx, \sin jx) = 0.$$

定义 6 设 $\varphi_n(x)$ 是 $[a,b]$ 上首项系数 $a_n\neq 0$ 的 n 次多项式，$\rho(x)$ 为 $[a,b]$ 上权函数，如果多项式序列 $\{\varphi_n(x)\}_0^\infty$ 满足关系式 (2.2)，则称多项式序列 $\{\varphi_n(x)\}_0^\infty$ 为在 $[a,b]$ 上带权 $\rho(x)$ 正交，称 $\varphi_n(x)$ 为 $[a,b]$ 上带权 $\rho(x)$ 的 n 次正交多项式.

只要给定区间 $[a,b]$ 及权函数 $\rho(x)$，均可由一族线性无关的幂函数 $\{1,x,\cdots,x^n,\cdots\}$，利用逐个正交化手续构造出正交多项式序列 $\{\varphi_n(x)\}_0^\infty$：

$$\varphi_0(x) = 1, \qquad \varphi_n(x) = x^n - \sum_{j=0}^{n-1} \frac{(x^n, \varphi_j(x))}{(\varphi_j(x), \varphi_j(x))}\varphi_j(x)$$

$$(n = 1,2,\cdots). \tag{2.3}$$

这样得到的正交多项式序列有以下性质：

(1) $\varphi_n(x)$ 是具有最高次项系数为 1 的 n 次多项式.

(2) 任何 n 次多项式 $P_n(x)\in H_n$ 均可表示为 $\varphi_0(x)$，$\varphi_1(x),\cdots,\varphi_n(x)$ 的线性组合.

(3) 当 $k\neq j$ 时，$(\varphi_j(x),\varphi_k(x))=0$，且 $\varphi_k(x)$ 与任一次数小于 k 的多项式正交.

(4) 成立递推关系

$$\varphi_{n+1}(x) = (x-\alpha_n)\varphi_n(x) - \beta_n\varphi_{n-1}(x) \qquad (n = 0,1,\cdots), \tag{2.4}$$

其中

$$\varphi_0(x) = 1, \qquad \varphi_{-1}(x) = 0,$$

$$\alpha_n = (x\varphi_n(x), \varphi_n(x))/(\varphi_n(x), \varphi_n(x)),$$

$$\beta_n = (\varphi_n(x), \varphi_n(x))/(\varphi_{n-1}(x), \varphi_{n-1}(x)) \qquad (n = 1,2,\cdots),$$

这里 $(x\varphi_n(x),\varphi_n(x))=\int_a^b x\varphi_n^2(x)\rho(x)\mathrm{d}x$.

(5) 设$\{\varphi_n(x)\}_0^\infty$ 是在$[a,b]$上带权 $\rho(x)$的正交多项式序列，则 $\varphi_n(x)(n\geqslant 1)$的 n 个根都是在区间(a,b)内的单重实根.

以上性质证明可见[2]，下面给出几种常见的正交多项式.

3.2.2 勒让德多项式

当区间为$[-1,1]$，权函数 $\rho(x)\equiv 1$ 时，由$\{1,x,\cdots,x^n,\cdots\}$正交化得到的多项式就称为**勒让德**(Legendre)**多项式**，并用 $\mathrm{P}_0(x)$，$\mathrm{P}_1(x)$，…，$\mathrm{P}_n(x)$，…表示. 这是勒让德于 1785 年引进的. 1814 年**罗德利克**(Rodrigul)给出了简单的表达式

$$\mathrm{P}_0(x)=1,\quad \mathrm{P}_n(x)=\frac{1}{2^n n!}\frac{\mathrm{d}^n}{\mathrm{d}x^n}\{(x^2-1)^n\}\quad (n=1,2,\cdots),\tag{2.5}$$

由于$(x^2-1)^n$ 是 $2n$ 次多项式，求 n 阶导数后得

$$\mathrm{P}_n(x)=\frac{1}{2^n n!}(2n)(2n-1)\cdots(n+1)x^n+a_{n-1}x^{n-1}+\cdots+a_0,$$

于是得首项 x^n 的系数 $a_n=\frac{(2n)!}{2^n(n!)^2}$. 显然最高项系数为 1 的勒让德多项式为

$$\widetilde{\mathrm{P}}_n(x)=\frac{n!}{(2n)!}\frac{\mathrm{d}^n}{\mathrm{d}x^n}[(x^2-1)^n].\tag{2.6}$$

勒让德多项式有下述几个重要性质：

性质 1 正交性

$$\int_{-1}^1 \mathrm{P}_n(x)\mathrm{P}_m(x)\mathrm{d}x=\begin{cases}0, & m\neq n;\\ \dfrac{2}{2n+1}, & m=n.\end{cases}\tag{2.7}$$

证明 令 $\varphi(x)=(x^2-1)^n$，则

$$\varphi^{(k)}(\pm 1)=0\qquad (k=0,1,\cdots,n-1).$$

设 $Q(x)$ 是在区间 $[-1,1]$ 上有 n 阶连续可微的函数，由分部积分知

$$\begin{aligned}\int_{-1}^{1} P_n(x)Q(x)\mathrm{d}x &= \frac{1}{2^n n!}\int_{-1}^{1} Q(x)\varphi^{(n)}(x)\mathrm{d}x \\ &= -\frac{1}{2^n n!}\int_{-1}^{1} Q'(x)\varphi^{(n-1)}(x)\mathrm{d}x \\ &= \cdots \\ &= \frac{(-1)^n}{2^n n!}\int_{-1}^{1} Q^{(n)}(x)\varphi(x)\mathrm{d}x.\end{aligned}$$

下面分两种情况讨论.

(1) 若 $Q(x)$ 是次数小于 n 的多项式，则 $Q^{(n)}(x)\equiv 0$，故得

$$\int_{-1}^{1} P_m(x)P_n(x)\mathrm{d}x = 0,\quad \text{当 } n \neq m.$$

(2) 若 $Q(x)=P_n(x)=\dfrac{1}{2^n n!}\varphi^{(n)}(x)=\dfrac{(2n)!}{2^n(n!)^2}x^n+\cdots$,

$$Q^{(n)}(x) = P_n^{(n)}(x) = \frac{(2n)!}{2^n n!},$$

于是

$$\begin{aligned}\int_{-1}^{1} P_n^2(x)\mathrm{d}x &= \frac{(-1)^n(2n)!}{2^{2n}(n!)^2}\int_{-1}^{1}(x^2-1)^n\mathrm{d}x \\ &= \frac{(2n)!}{2^{2n}(n!)^2}\int_{-1}^{1}(1-x^2)^n\mathrm{d}x.\end{aligned}$$

由于

$$\int_0^1 (1-x^2)^n\mathrm{d}x = \int_0^{\frac{\pi}{2}}\cos^{2n+1}t\,\mathrm{d}t = \frac{2\cdot 4\cdot\cdots\cdot(2n)}{1\cdot 3\cdot\cdots\cdot(2n+1)},$$

故

$$\int_{-1}^{1} P_n^2(x)\mathrm{d}x = \frac{2}{2n+1},$$

于是(2.7)得证.

性质 2　奇偶性

$$P_n(-x) = (-1)^n P_n(x). \tag{2.8}$$

由于 $\varphi(x)=(x^2-1)^n$ 是偶次多项式，经过偶次求导仍为偶次多项式，经过奇次求导则为奇次多项式，故 n 为偶数时 $P_n(x)$ 为偶函数，n 为奇数时 $P_n(x)$ 为奇函数，于是(2.8)成立.

性质 3 递推关系

考虑 $n+1$ 次多项式 $xP_n(x)$，它可表示为

$$xP_n(x) = a_0P_0(x) + a_1P_1(x) + \cdots + a_{n+1}P_{n+1}(x).$$

两边乘 $P_k(x)$，并从 -1 到 1 积分，得

$$\int_{-1}^{1} xP_n(x)P_k(x)\mathrm{d}x = a_k\int_{-1}^{1} P_k^2(x)\mathrm{d}x.$$

当 $k\leqslant n-2$ 时，$xP_k(x)$ 次数小于等于 $n-1$，上式左端积分为 0，故得 $a_k=0$. 当 $k=n$ 时，$xP_n^2(x)$ 为奇函数，左端积分仍为 0，故 $a_n=0$. 于是

$$xP_n(x) = a_{n-1}P_{n-1}(x) + a_{n+1}P_{n+1}(x),$$

其中

$$a_{n-1} = \frac{2n-1}{2}\int_{-1}^{1} xP_n(x)P_{n-1}(x)\mathrm{d}x = \frac{2n-1}{2}\cdot\frac{2n}{4n^2-1} = \frac{n}{2n+1},$$

$$a_{n+1} = \frac{2n+3}{2}\int_{-1}^{1} xP_n(x)P_{n+1}(x)\mathrm{d}x = \frac{2n+3}{2}\cdot\frac{2(n+1)}{(2n+1)(2n+3)} = \frac{n+1}{2n+1},$$

从而得到以下的递推公式

$$(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x) \quad (n=1,2,\cdots), \tag{2.9}$$

由 $P_0(x)=1, P_1(x)=x$，利用(2.9)就可推出

$$P_2(x) = (3x^2-1)/2,$$

$$P_3(x) = (5x^3-3x)/2,$$

$$P_4(x) = (35x^4-30x^2+3)/8,$$

$$P_5(x) = (63x^5 - 70x^3 - 15x)/8,$$

$$P_6(x) = (231x^6 - 315x^4 + 105x^2 - 5)/16,$$

$$\cdots$$

图 3-1 给出了 $P_0(x)$,$P_1(x)$,$P_2(x)$,$P_3(x)$的图形.

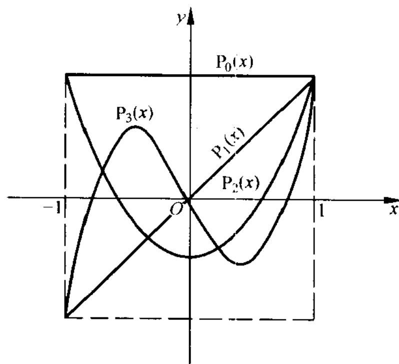


图 3-1

性质 4　$P_n(x)$在区间$[-1,1]$内有 n 个不同的实零点.

3.2.3　切比雪夫多项式

当权函数 $\rho(x) = \dfrac{1}{\sqrt{1-x^2}}$,区间为$[-1,1]$时,由序列$\{1, x, \cdots, x^n, \cdots\}$正交化得到的正交多项式就是**切比雪夫(Chebyshev)多项式**,它可表示为

$$T_n(x) = \cos(n\arccos x), \qquad |x| \leqslant 1. \tag{2.10}$$

若令 $x=\cos\theta$,则 $T_n(x)=\cos n\theta, 0\leqslant\theta\leqslant\pi$.

切比雪夫多项式有很多重要性质:

性质 5　递推关系

$$\begin{aligned} &T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x) \quad (n = 1,2,\cdots), \\ &T_0(x) = 1, \quad T_1(x) = x. \end{aligned} \tag{2.11}$$

这只要由三角恒等式

$$\cos(n+1)\theta = 2\cos\theta\cos n\theta - \cos(n-1)\theta \quad (n \geqslant 1)$$

令 $x=\cos\theta$ 即得. 由(2.11)就可推出

$$\begin{aligned}
T_2(x) &= 2x^2 - 1, \\
T_3(x) &= 4x^3 - 3x, \\
T_4(x) &= 8x^4 - 8x^2 + 1, \\
T_5(x) &= 16x^5 - 20x^3 + 5x, \\
T_6(x) &= 32x^6 - 48x^4 + 18x^2 - 1, \\
&\cdots
\end{aligned}$$

$T_0(x)$,$T_1(x)$,$T_2(x)$,$T_3(x)$的函数图形见图 3-2.

由递推关系(2.11)还可得到 $T_n(x)$的最高次项系数是 2^{n-1} $(n\geqslant 1)$.

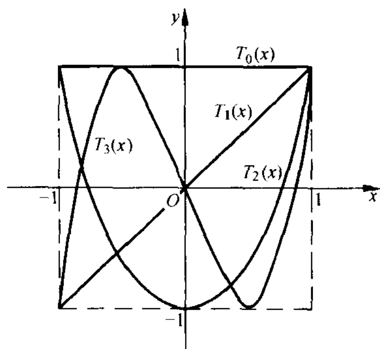


图 3-2

性质 6 切比雪夫多项式$\{T_k(x)\}$在区间$[-1,1]$上带权 $\rho(x)=1/\sqrt{1-x^2}$正交,且

$$\int_{-1}^{1} \frac{T_n(x)T_m(x)\,\mathrm{d}x}{\sqrt{1-x^2}} = \begin{cases} 0, & n \neq m; \\ \dfrac{\pi}{2}, & n = m \neq 0; \\ \pi, & n = m = 0. \end{cases} \tag{2.12}$$

事实上，令 $x=\cos\theta$，则 $dx=-\sin\theta d\theta$，于是

$$\int_{-1}^{1}\frac{T_n(x)T_m(x)}{\sqrt{1-x^2}}dx=\int_0^{\pi}\cos n\theta\cos m\theta d\theta=\begin{cases}0, & n\neq m;\\ \frac{\pi}{2}, & n=m\neq 0;\\ \pi, & n=m=0.\end{cases}$$

性质7 $T_{2k}(x)$ 只含 x 的偶次幂，$T_{2k+1}(x)$ 只含 x 的奇次幂.

这性质由递推关系直接得到.

性质8 $T_n(x)$ 在区间 $[-1,1]$ 上有 n 个零点

$$x_k=\cos\frac{2k-1}{2n}\pi,\quad k=1,\cdots,n.$$

此外，实际计算中时常要求 x^n 用 $T_0(x),T_1(x),\cdots,T_n(x)$ 的线性组合表示，其公式为

$$x^n=2^{1-n}\sum_{k=0}^{\left[\frac{n}{2}\right]}\binom{n}{k}T_{n-2k}(x).\tag{2.13}$$

这里规定 $T_0(x)=1$. $n=1,2,\cdots,6$ 时的结果如下：

$$1=T_0(x),$$

$$x=T_1(x),$$

$$x^2=\frac{1}{2}(T_0(x)+T_2(x)),$$

$$x^3=\frac{1}{4}(3T_1(x)+T_3(x)),$$

$$x^4=\frac{1}{8}(3T_0(x)+4T_2(x)+T_4(x)),$$

$$x^5=\frac{1}{16}(10T_1(x)+5T_3(x)+T_5(x)),$$

$$x^6=\frac{1}{32}(10T_0(x)+15T_2(x)+6T_4(x)+T_6(x)).$$

3.2.4 其他常用的正交多项式

一般说，如果区间$[a,b]$及权函数$\rho(x)$不同，则得到的正交多项式也不同. 除上述两种最重要的正交多项式外，下面再给出三种较常用的正交多项式.

1. 第二类切比雪夫多项式

在区间$[-1,1]$上带权$\rho(x)=\sqrt{1-x^2}$的正交多项式称为**第二类切比雪夫多项式**，其表达式为

$$U_n(x)=\frac{\sin[(n+1)\arccos x]}{\sqrt{1-x^2}}. \tag{2.14}$$

令$x=\cos\theta$，可得

$$\int_{-1}^{1}U_n(x)U_m(x)\sqrt{1-x^2}\,dx=\int_0^{\pi}\sin(n+1)\theta\sin(m+1)\theta d\theta$$

$$=\begin{cases}0, & m\neq n;\\ \dfrac{\pi}{2}, & m=n,\end{cases}$$

即$\{U_n(x)\}$是$[-1,1]$上带权$\sqrt{1-x^2}$的正交多项式族. 还可得到递推关系式

$$U_0(x)=1,\qquad U_1(x)=2x,$$

$$U_{n+1}(x)=2xU_n(x)-U_{n-1}(x)\quad(n=1,2,\cdots).$$

2. 拉盖尔多项式

在区间$[0,+\infty)$上带权e^{-x}的正交多项式称为**拉盖尔(Laguerre)多项式**，其表达式为

$$L_n(x)=e^x\frac{d^n}{dx^n}(x^ne^{-x}). \tag{2.15}$$

它也具有正交性质

$$\int_0^{\infty}e^{-x}L_n(x)L_m(x)dx=\begin{cases}0, & m\neq n;\\ (n!)^2, & m=n,\end{cases}$$

和递推关系

$$L_0(x)=1,\qquad L_1(x)=1-x,$$

$$L_{n+1}(x)=(1+2n-x)L_n(x)-n^2L_{n-1}(x)(n=1,2,\cdots).$$

3. 埃尔米特多项式

在区间$(-\infty,+\infty)$上带权 e^{-x^2} 的正交多项式称为**埃尔米特多项式**,其表达式为

$$H_n(x)=(-1)^n e^{x^2}\frac{d^n}{dx^n}(e^{-x^2}),\tag{2.16}$$

它满足正交关系

$$\int_{-\infty}^{+\infty}e^{-x^2}H_m(x)H_n(x)dx=\begin{cases}0, & m\neq n;\\ 2^n n!\sqrt{\pi}, & m=n,\end{cases}$$

并有递推关系

$$H_0(x)=1,\qquad H_1(x)=2x,$$

$$H_{n+1}(x)=2xH_n(x)-2nH_{n-1}(x)\quad(n=1,2,\cdots).$$

3.3 最佳一致逼近多项式

3.3.1 基本概念及其理论

本节讨论 $f\in C[a,b]$,在 $H_n=\mathrm{span}\{1,x,\cdots,x^n\}$中求多项式 $P_n^*(x)$,使其误差

$$\|f-P_n^*\|_\infty=\max_{a\leqslant x\leqslant b}|f(x)-P_n^*(x)|=\min_{P_n\in H_n}\|f-P_n\|.$$

这就是通常所谓最佳一致逼近或切比雪夫逼近问题. 为了说明这一概念,先给出以下定义.

定义 7 设 $P_n(x)\in H_n, f(x)\in C[a,b]$,称

$$\Delta(f,P_n)=\|f-P_n\|_\infty=\max_{a\leqslant x\leqslant b}|f(x)-P_n(x)|\tag{3.1}$$

为 $f(x)$与 $P_n(x)$在$[a,b]$上的**偏差**.

显然 $\Delta(f,P_n)\geqslant 0$,$\Delta(f,P_n)$的全体组成一个集合,记为

$\{\Delta(f,P_n)\}$,它有下界 0. 若记集合的下确界为

$$E_n = \inf_{P_n \in H_n} \{\Delta(f,P_n)\} = \inf_{P_n \in H_n} \max_{a \leqslant x \leqslant b} | f(x) - P_n(x) |, \tag{3.2}$$

则称之为 $f(x)$在$[a,b]$上的**最小偏差**.

定义 8 假定 $f(x) \in C[a,b]$,若存在 $P_n^*(x) \in H_n$ 使得

$$\Delta(f,P_n^*) = E_n, \tag{3.3}$$

则称 $P_n^*(x)$是 $f(x)$在$[a,b]$上的**最佳一致逼近多项式**或**最小偏差逼近多项式**,简称**最佳逼近多项式**.

注意,定义并未说明最佳逼近多项式是否存在,但可证明下面的存在定理.

定理 4 若 $f(x) \in C[a,b]$,则总存在 $P_n^*(x) \in H_n$,使

$$\| f(x) - P_n^*(x) \|_\infty = E_n.$$

证明略,可参考[2].

为了研究最佳逼近多项式的特性,先引进偏差点的定义.

定义 9 设 $f(x) \in C[a,b]$,$P(x) \in H_n$,若在 $x=x_0$ 上有

$$| P(x_0) - f(x_0) | = \max_{a \leqslant x \leqslant b} | P(x) - f(x) | = \mu,$$

就称 x_0 是 $P(x)$的**偏差点**.

若 $P(x_0) - f(x_0) = \mu$,称 x_0 为"**正**"**偏差点**.

若 $P(x_0) - f(x_0) = -\mu$,称 x_0 为"**负**"**偏差点**.

由于函数 $P(x) - f(x)$在$[a,b]$上连续,因此,至少存在一个点 $x_0 \in [a,b]$,使

$$| P(x_0) - f(x_0) | = \mu,$$

也就是说 $P(x)$的偏差点总是存在的. 下面给出反映最佳逼近多项式特征的切比雪夫定理.

定理 5 $P(x) \in H_n$ 是 $f(x) \in C[a,b]$的最佳逼近多项式的充分必要条件是 $P(x)$在$[a,b]$上至少有 $n+2$ 个轮流为"正"、"负"的偏差点,即有 $n+2$ 个点 $a \leqslant x_1 < x_2 < \cdots < x_{n+2} \leqslant b$,使

$$P(x_k) - f(x_k) = (-1)^k \sigma \| P(x) - f(x) \|_\infty, \ \sigma = \pm 1, \tag{3.4}$$

这样的点组称为**切比雪夫交错点组**.

证明 只证充分性. 假定在$[a,b]$上有 $n+2$ 个点使(3.4)成立,要证明 $P(x)$是 $f(x)$在$[a,b]$上的最佳逼近多项式. 用反证法,若存在 $Q(x)\in H_n$,$Q(x)\neq P(x)$,使

$$\| f(x) - Q(x) \|_\infty < \| f(x) - P(x) \|_\infty.$$

由于

$$P(x) - Q(x) = [P(x) - f(x)] - [Q(x) - f(x)]$$

在点 $x_1, x_2, \cdots, x_{n+2}$上的符号与 $P(x_k) - f(x_k)(k=1,\cdots,n+2)$一致,故 $P(x)-Q(x)$也在 $n+2$ 个点上轮流取"+"、"−"号. 由连续函数性质,它在$[a,b]$内有 $n+1$ 个零点,但因 $P(x)-Q(x)\neq 0$是不超过 n 次的多项式,它的零点不超过 n. 这矛盾说明假设不对,故 $P(x)$就是所求最佳逼近多项式. 充分性得证. 必要性证明略,可参看[5].

定理 5 说明用 $P(x)$逼近 $f(x)$的误差曲线 $y=P(x)-f(x)$是均匀分布的. 由这定理还可得以下重要推论.

推论 1 若 $f(x)\in C[a,b]$,则在 H_n 中存在唯一的最佳逼近多项式.

证明略.

利用定理 5 可直接得到切比雪夫多项式 $T_n(x)$的一个重要性质,即

定理 6 在区间$[-1,1]$上所有最高次项系数为 1 的 n 次多项式中,$\omega_n(x)=\frac{1}{2^{n-1}}T_n(x)$与零的偏差最小,其偏差为$\frac{1}{2^{n-1}}$.

证明 由于

$$\omega_n(x) = \frac{1}{2^{n-1}} T_n(x) = x^n - P_{n-1}^*(x),$$

$$\max_{-1\leqslant x\leqslant 1}|\omega_n(x)|=\frac{1}{2^{n-1}}\cdot\max_{-1\leqslant x\leqslant 1}|T_n(x)|=\frac{1}{2^{n-1}},$$

且点 $x_k=\cos\frac{k}{n}\pi(k=0,1,\cdots,n)$是 $T_n(x)$的切比雪夫交错点组，由定理 5 可知，区间$[-1,1]$上 x^n 在 H_{n-1}中最佳逼近多项式为 $P_{n-1}^*(x)$，即 $\omega_n(x)$是与零的偏差最小的多项式. 定理得证.

例 3 求 $f(x)=2x^3+x^2+2x-1$ 在$[-1,1]$上的最佳 2 次逼近多项式.

解 由题意，所求最佳逼近多项式 $P_2^*(x)$应满足

$$\max_{-1\leqslant x\leqslant 1}|f(x)-P_2^*(x)|=\min.$$

由定理 6 可知，当

$$f(x)-P_2^*(x)=\frac{1}{2}T_3(x)=2x^3-\frac{3}{2}x$$

时，多项式 $f(x)-P_2^*(x)$与零偏差最小，故

$$P_2^*(x)=f(x)-\frac{1}{2}T_3(x)$$

$$=x^2+\frac{7}{2}x-1$$

就是 $f(x)$在$[-1,1]$上的最佳 2 次逼近多项式.

3.3.2 最佳一次逼近多项式

定理 5 给出了最佳逼近多项式 $P(x)$的特性，但要求出 $P(x)$却相当困难. 下面讨论 $n=1$ 的情形. 假定 $f(x)\in C^2[a,b]$，且 $f''(x)$在(a,b)内不变号，我们要求最佳一次逼近多项式 $P_1(x)=a_0+a_1x$. 根据定理 5 可知至少有 3 个点 $a\leqslant x_1<x_2<x_3\leqslant b$，使

$$P_1(x_k)-f(x_k)=(-1)^k\sigma\max_{a\leqslant x\leqslant b}|P_1(x)-f(x)|$$

$$(\sigma=\pm 1,k=1,2,3).$$

由于 $f''(x)$在$[a,b]$上不变号，故 $f'(x)$单调，$f'(x)-a_1$ 在

(a,b)内只有一个零点，记为 x_2，于是

$$P_1'(x_2)-f'(x_2)=a_1-f'(x_2)=0,\text{即 } f'(x_2)=a_1.$$

另外两个偏差点必是区间端点，即 $x_1=a,x_2=b$，且满足

$$P_1(a)-f(a)=P_1(b)-f(b)=-[P_1(x_2)-f(x_2)].$$

由此得到

$$\begin{cases}a_0+a_1a-f(a)=a_0+a_1b-f(b);\\ a_0+a_1a-f(a)=f(x_2)-(a_0+a_1x_2).\end{cases}\tag{3.5}$$

解出

$$a_1=\frac{f(b)-f(a)}{b-a}=f'(x_2),\tag{3.6}$$

代入(3.5)得

$$a_0=\frac{f(a)+f(x_2)}{2}-\frac{f(b)-f(a)}{b-a}\,\frac{a+x_2}{2}.\tag{3.7}$$

这就得到最佳一次逼近多项式 $P_1(x)$，其几何意义如图 3-3 所示. 直线 $y=P_1(x)$与弦 MN 平行，且通过 MQ 的中点 D，其方程为

$$y=\frac{1}{2}[f(a)+f(x_2)]+a_1\left(x-\frac{a+x_2}{2}\right).$$

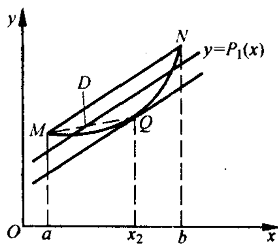


图　3-3

例 4　求 $f(x)=\sqrt{1+x^2}$在$[0,1]$上的最佳一次逼近多项式.

解　由(3.6)可算出

$$a_1=\sqrt{2}-1\approx 0.414,$$

又 $f'(x)=\dfrac{x}{\sqrt{1+x^2}}$，故$\dfrac{x_2}{\sqrt{1+x_2^2}}=\sqrt{2}-1$，解得

$$x_2=\sqrt{\frac{\sqrt{2}-1}{2}}\approx 0.4551, f(x_2)=\sqrt{1+x_2^2}\approx 1.0986.$$

由(3.7)，得

$$a_0=\frac{1+\sqrt{1+x_2^2}}{2}-a_1\frac{x_2}{2}\approx 0.955,$$

于是得$\sqrt{1+x^2}$的最佳一次逼近多项式为

$$P_1(x)=0.955+0.414x,$$

即

$$\sqrt{1+x^2}\approx 0.955+0.414x, 0\leqslant x\leqslant 1; \tag{3.8}$$

误差限为

$$\max_{0\leqslant x\leqslant 1}|\sqrt{1+x^2}-P_1(x)|\leqslant 0.045.$$

在(3.8)中若令 $x=\dfrac{b}{a}\leqslant 1$，则可得一个求根式的公式

$$\sqrt{a^2+b^2}\approx 0.955a+0.414b.$$

3.4 最佳平方逼近

3.4.1 最佳平方逼近及其计算

现在我们研究在区间$[a,b]$上一般的最佳平方逼近问题. 对 $f(x)\in C[a,b]$ 及 $C[a,b]$ 中的一个子集 $\varphi=\mathrm{span}\{\varphi_0(x),\varphi_1(x),\cdots,\varphi_n(x)\}$，若存在 $S^*(x)\in\varphi$，使

$$\|f(x)-S^*(x)\|_2^2=\min_{S(x)\in\varphi}\|f(x)-S(x)\|_2^2$$

$$=\min_{S(x)\in\varphi}\int_a^b\rho(x)[f(x)-S(x)]^2\mathrm{d}x. \tag{4.1}$$

则称 $S^*(x)$是 $f(x)$在子集 $\varphi\subset C[a,b]$中的**最佳平方逼近函数**. 为了求 $S^*(x)$，由(4.1)可知该问题等价于求多元函数

$$I(a_0,a_1,\cdots,a_n)=\int_a^b \rho(x)\Big[\sum_{j=0}^{n}a_j\varphi_j(x)-f(x)\Big]^2\mathrm{d}x \tag{4.2}$$

的最小值. 由于 $I(a_0,a_1,\cdots,a_n)$ 是关于 $a_0,a_1,\cdots,a_n$ 的二次函数，利用多元函数求极值的必要条件

$$\frac{\partial I}{\partial a_k}=0 \qquad (k=0,1,\cdots,n),$$

即

$$\frac{\partial I}{\partial a_k}=2\int_a^b \rho(x)\Big[\sum_{j=0}^{n}a_j\varphi_j(x)-f(x)\Big]\varphi_k(x)\mathrm{d}x=0$$

$$(k=0,1,\cdots,n),$$

于是有

$$\sum_{j=0}^{n}(\varphi_k(x),\varphi_j(x))a_j=(f(x),\varphi_k(x)) \qquad (k=0,1,\cdots,n). \tag{4.3}$$

这是关于 $a_0,a_1,\cdots,a_n$ 的线性方程组，称为**法方程**，由于 $\varphi_0(x)$，$\varphi_1(x),\cdots,\varphi_n(x)$ 线性无关，故系数 $\det\boldsymbol{G}(\varphi_0,\varphi_1,\cdots,\varphi_n)\neq 0$，于是方程组(4.3)有唯一解 $a_k=a_k^*$ $(k=0,1,\cdots,n)$，从而得到

$$S^*(x)=a_0^*\varphi_0(x)+\cdots+a_n^*\varphi_n(x).$$

下面证明 $S^*(x)$ 满足(4.1)，即对任何 $S(x)\in\varphi$，有

$$\int_a^b \rho(x)[f(x)-S^*(x)]^2\mathrm{d}x\leqslant\int_a^b \rho(x)[f(x)-S(x)]^2\mathrm{d}x. \tag{4.4}$$

为此只要考虑

$$\begin{aligned}D=&\int_a^b \rho(x)[f(x)-S(x)]^2\mathrm{d}x-\int_a^b \rho(x)[f(x)-S^*(x)]^2\mathrm{d}x\\=&\int_a^b \rho(x)[S(x)-S^*(x)]^2\mathrm{d}x\\&+2\int_a^b \rho(x)[S^*(x)-S(x)][f(x)-S^*(x)]\mathrm{d}x.\end{aligned}$$

由于 $S^*(x)$ 的系数 a_k^* 是方程(4.3)的解，故

$$\int_a^b \rho(x)[f(x)-S^*(x)]\varphi_k(x)\mathrm{d}x = 0 \qquad (k=0,1,\cdots,n),$$

从而上式第二个积分为 0，于是

$$D=\int_a^b \rho(x)[S(x)-S^*(x)]^2\mathrm{d}x \geqslant 0,$$

故(4.4)成立. 这就证明了 $S^*(x)$ 是 $f(x)$ 在 φ 中的最佳平方逼近函数.

若令 $\delta(x)=f(x)-S^*(x)$，则平方误差为

$$\begin{aligned}\|\delta(x)\|_2^2 &= (f(x)-S^*(x),f(x)-S^*(x))\\ &=(f(x),f(x))-(S^*(x),f(x))\\ &=\|f(x)\|_2^2-\sum_{k=0}^n a_k^*(\varphi_k(x),f(x)).\end{aligned} \tag{4.5}$$

若取 $\varphi_k(x)=x^k,\rho(x)\equiv 1,f(x)\in C[0,1]$，则要在 H_n 中求 n 次最佳平方逼近多项式

$$S^*(x)=a_0^*+a_1^*x+\cdots+a_n^*x^n,$$

此时

$$(\varphi_j(x),\varphi_k(x))=\int_0^1 x^{k+j}\mathrm{d}x=\frac{1}{k+j+1},$$

$$(f(x),\varphi_k(x))=\int_0^1 f(x)x^k\mathrm{d}x\equiv d_k.$$

若用 $\boldsymbol{H}$ 表示 $\boldsymbol{G}_n=\boldsymbol{G}(1,x,\cdots,x^n)$ 对应的矩阵，即

$$\boldsymbol{H}=\begin{pmatrix}1 & 1/2 & \cdots & 1/(n+1)\\ 1/2 & 1/3 & \cdots & 1/(n+2)\\ \vdots & \vdots & & \vdots\\ 1/(n+1) & 1/(n+2) & \cdots & 1/(2n+1)\end{pmatrix} \tag{4.6}$$

称为希尔伯特(Hilbert)矩阵，记 $\boldsymbol{a}=(a_0,a_1,\cdots,a_n)^{\mathrm{T}}$，$\boldsymbol{d}=(d_0,d_1,\cdots,d_n)^{\mathrm{T}}$，则

$$\boldsymbol{Ha}=\boldsymbol{d} \tag{4.7}$$

的解 $a_k=a_k^*$ $(k=0,1,\cdots,n)$即为所求.

例 5　设 $f(x)=\sqrt{1+x^2}$，求$[0,1]$上的一次最佳平方逼近多项式.

解　利用(4.7)，得

$$d_0=\int_0^1\sqrt{1+x^2}\mathrm{d}x=\frac{1}{2}\ln(1+\sqrt{2})+\frac{\sqrt{2}}{2}\approx 1.147,$$

$$d_1=\int_0^1 x\sqrt{1+x^2}\mathrm{d}x=\frac{1}{3}(1+x^2)^{3/2}\Big|_0^1=\frac{2\sqrt{2}-1}{3}\approx 0.609,$$

得方程组

$$\begin{pmatrix}1 & \frac{1}{2}\\ \frac{1}{2} & \frac{1}{3}\end{pmatrix}\begin{pmatrix}a_0\\ a_1\end{pmatrix}=\begin{pmatrix}1.147\\ 0.609\end{pmatrix},$$

解出

$$a_0=0.934,\quad a_1=0.426,$$

故

$$S_1^*(x)=0.934+0.426x.$$

平方误差

$$\begin{aligned}\|\delta(x)\|_2^2&=(f(x),f(x))-(S_1^*(x),f(x))\\&=\int_0^1(1+x^2)\mathrm{d}x-0.426d_1-0.934d_0=0.0026.\end{aligned}$$

最大误差

$$\|\delta(x)\|_\infty=\max_{0\leqslant x\leqslant 1}\left|\sqrt{1+x^2}-S_1^*(x)\right|\approx 0.066.$$

用$\{1,x,\cdots,x^n\}$做基，求最佳平方逼近多项式，当 n 较大时，系数矩阵(4.6)是高度病态的(见第 5 章)，因此直接求解法方程是相当困难的，通常是采用正交多项式做基.

3.4.2 用正交函数族作最佳平方逼近

设 $f(x)\in C[a,b]$，$\varphi=\mathrm{span}\{\varphi_0(x),\varphi_1(x),\cdots,\varphi_n(x)\}$，若 $\varphi_0(x),\varphi_1(x),\cdots,\varphi_n(x)$ 是满足条件(2.2)的正交函数族，则 $(\varphi_i(x),\varphi_j(x))=0$，$i\neq j$ 而 $(\varphi_j(x),\varphi_j(x))>0$，故法方程(4.3)的系数矩阵 $\boldsymbol{G}_n=\boldsymbol{G}(\varphi_0(x),\varphi_1(x),\cdots,\varphi_n(x))$ 为非奇异对角阵，且方程(4.3)的解为

$$a_k^*=(f(x),\varphi_k(x))/(\varphi_k(x),\varphi_k(x))\qquad(k=0,1,\cdots,n).\tag{4.8}$$

于是 $f(x)\in C[a,b]$ 在 φ 中的最佳平方逼近函数为

$$S^*(x)=\sum_{k=0}^{n}\frac{(f(x),\varphi_k(x))}{\|\varphi_k(x)\|_2^2}\varphi_k(x).\tag{4.9}$$

由(4.5)可得均方误差为

$$\begin{aligned}\|\delta_n(x)\|_2&=\|f(x)-S_n^*(x)\|_2\\&=\left(\|f(x)\|_2^2-\sum_{k=0}^{n}\left[\frac{(f(x),\varphi_k(x))}{\|\varphi_k(x)\|_2}\right]^2\right)^{\frac{1}{2}}.\end{aligned}\tag{4.10}$$

由此可得**贝塞尔(Bessel)不等式**

$$\sum_{k=1}^{n}(a_k^*\|\varphi_k(x)\|_2)^2\leqslant\|f(x)\|_2^2.\tag{4.11}$$

若 $f(x)\in C[a,b]$，按正交函数族 $\{\varphi_k(x)\}$ 展开，系数 a_k^* $(k=0,1,\cdots)$ 按(4.8)计算，得级数

$$\sum_{k=0}^{\infty}a_k^*\varphi_k(x)\tag{4.12}$$

称为 $f(x)$ 的**广义傅里叶(Foureir)级数**，系数 a_k^* 称为广义傅里叶系数. 它是傅里叶级数的直接推广.

下面讨论特殊情况，设 $\{\varphi_0(x),\varphi_1(x),\cdots,\varphi_n(x)\}$ 是正交多项式，$\varphi=\mathrm{span}\{\varphi_0(x),\varphi_1(x),\cdots,\varphi_n(x)\}$，$\varphi_k(x)$ $(k=0,1,\cdots,n)$ 可由 $1,x,\cdots,x^n$ 正交化得到，则有下面的收敛定理.

定理 7　设 $f(x)\in C[a,b]$，$S^*(x)$是由(4.9)给出的 $f(x)$的最佳平方逼近多项式，其中$\{\varphi_k(x),k=0,1,\cdots,n\}$是正交多项式族，则有

$$\lim_{n\to\infty}\|f(x)-S_n^*(x)\|_2=0.$$

证明略，可见[6].

下面考虑函数 $f(x)\in C[-1,1]$，按勒让德多项式$\{P_0(x),P_1(x),\cdots,P_n(x)\}$展开，由(4.8)，(4.9)可得

$$S_n^*(x)=a_0^*P_0(x)+a_1^*P_1(x)+\cdots+a_n^*P_n(x),\tag{4.13}$$

其中

$$a_k^*=\frac{(f(x),P_k(x))}{(P_k(x),P_k(x))}=\frac{2k+1}{2}\int_{-1}^{1}f(x)P_k(x)\mathrm{d}x.\tag{4.14}$$

根据(4.10)，平方误差为

$$\|\delta_k(x)\|_2^2=\int_{-1}^{1}f^2(x)\mathrm{d}x-\sum_{k=0}^{n}\frac{2}{2k+1}a_k^{*2}.\tag{4.15}$$

由定理 7 可得

$$\lim_{n\to\infty}\|f(x)-S_n^*(x)\|_2=0.$$

如果 $f(x)$满足光滑性条件还可得到 $S_n^*(x)$一致收敛于 $f(x)$的结论.

定理 8　设 $f(x)\in C^2[-1,1]$，$S_n^*(x)$由(4.13)给出，则对任意 $x\in[-1,1]$和$\forall\varepsilon>0$，当 n 充分大时有

$$|f(x)-S_n^*(x)|\leqslant\frac{\varepsilon}{\sqrt{n}}.$$

证明可见[6].

对于首项系数为 1 的勒让德多项式 $\tilde{P}_n$(由公式(2.6)给出)有以下性质.

定理 9　在所有最高次项系数为 1 的 n 次多项式中，勒让德多项式 $\tilde{P}_n(x)$在$[-1,1]$上与零的平方误差最小.

证明 设 $Q_n(x)$ 是任意一个最高次项系数为 1 的 n 次多项式，它可表示为

$$Q_n(x) = \widetilde{P}_n(x) + \sum_{k=0}^{n-1} a_k \widetilde{P}_k(x),$$

于是

$$\begin{aligned}
\| Q_n(x) \|_2^2 &= (Q_n(x), Q_n(x)) = \int_{-1}^{1} Q_n^2(x) \mathrm{d}x \\
&= (\widetilde{P}_n(x), \widetilde{P}_n(x)) + \sum_{k=0}^{n-1} a_k^2 (\widetilde{P}_k(x), \widetilde{P}_k(x)) \\
&\geqslant (\widetilde{P}_n(x), \widetilde{P}_n(x)) \\
&= \| \widetilde{P}_n(x) \|_2^2.
\end{aligned}$$

当且仅当 $a_0 = a_1 = \cdots = a_{n-1} = 0$ 时等号才成立，即当 $Q_n(x) \equiv \widetilde{P}_n(x)$ 时平方误差最小.

例 6 求 $f(x) = \mathrm{e}^x$ 在 $[-1,1]$ 上的三次最佳平方逼近多项式.

解 先计算 $(f(x), \widetilde{P}_k(x))$ $(k=0,1,2,3)$.

$$(f(x), P_0(x)) = \int_{-1}^{1} \mathrm{e}^x \mathrm{d}x = \mathrm{e} - \frac{1}{\mathrm{e}} \approx 2.3504;$$

$$(f(x), P_1(x)) = \int_{-1}^{1} x\mathrm{e}^x \mathrm{d}x = 2\mathrm{e}^{-1} \approx 0.7358;$$

$$(f(x), P_2(x)) = \int_{-1}^{1} \left(\frac{3}{2}x^2 - \frac{1}{2}\right) \mathrm{e}^x \mathrm{d}x = \mathrm{e} - \frac{7}{\mathrm{e}} \approx 0.1431;$$

$$(f(x), P_3(x)) = \int_{-1}^{1} \left(\frac{5}{2}x^3 - \frac{3}{2}x\right) \mathrm{e}^x \mathrm{d}x = 37 \frac{1}{\mathrm{e}} - 5\mathrm{e} \approx 0.02013.$$

由 (4.14) 得

$$\begin{aligned}
a_0^* &= (f(x), P_0(x))/2 = 1.1752, \\
a_1^* &= 3(f(x), P_1(x))/2 = 1.1036, \\
a_2^* &= 5(f(x), P_2(x))/2 = 0.3578, \\
a_3^* &= 7(f(x), P_3(x))/2 = 0.07046.
\end{aligned}$$

代入(4.13)得

$$S_3^*(x) = 0.9963 + 0.9979x + 0.5367x^2 + 0.1761x^3.$$

均方误差

$$\| \delta_n(x) \|_2 = \| e^x - S_3^*(x) \|_2 = \sqrt{\int_{-1}^{1} e^{2x} dx - \sum_{k=0}^{3} \frac{2}{2k+1} a_k^{*2}} \leqslant 0.0084.$$

最大误差

$$\| \delta_n(x) \|_\infty = \| e^x - S_3(x) \|_\infty \leqslant 0.0112.$$

如果 $f(x) \in C[a,b]$，求 $[a,b]$ 上的最佳平方逼近多项式，做变换

$$x = \frac{b-a}{2}t + \frac{b+a}{2} \quad (-1 \leqslant t \leqslant 1),$$

于是 $F(t) = f\left(\frac{b-a}{2}t + \frac{b+a}{2}\right)$ 在 $[-1,1]$ 上可用勒让德多项式做最佳平方逼近多项式 $S_n^*(t)$，从而得到区间 $[a,b]$ 上的最佳平方逼近多项式 $S_n^*\left(\frac{1}{b-a}(2x-a-b)\right)$.

由于勒让德多项式 $\{P_k(x)\}$ 是在区间 $[-1,1]$ 上由 $\{1, x, \cdots, x^k, \cdots\}$ 正交化得到的，因此利用函数的勒让德展开部分和得到最佳平方逼近多项式与由

$$S^*(x) = a_0 + a_1 x + \cdots + a_n x^n$$

直接通过解法方程得到 H_n 中的最佳平方逼近多项式是一致的，只是当 n 较大时法方程出现病态，计算误差较大，不能使用，而用勒让德展开不用解线性方程组，不存在病态问题，计算公式比较方便，因此通常都用这种方法求最佳平方逼近多项式.

3.5 曲线拟合的最小二乘法

3.5.1 最小二乘法及其计算

在函数的最佳平方逼近中 $f(x) \in C[a,b]$，如果 $f(x)$ 只在一

组离散点集$\{x_i, i=0,1,\cdots,m\}$上给定，这就是科学实验中经常见到的实验数据$\{(x_i, y_i), i=0,1,\cdots,m\}$的曲线拟合，这里$y_i=f(x_i), i=0,1,\cdots,m$，要求一个函数$y=S^*(x)$与所给数据$\{(x_i, y_i), i=0,1,\cdots,m\}$拟合，若记误差$\delta_i=S^*(x_i)-y_i, i=0,1,\cdots,m$，$\boldsymbol{\delta}=(\delta_0,\delta_1,\cdots,\delta_m)^{\mathrm{T}}$，设$\varphi_0(x),\varphi_1(x),\cdots,\varphi_n(x)$是$C[a,b]$上线性无关函数族，在$\varphi=\mathrm{span}\{\varphi_0(x),\varphi_1(x),\cdots,\varphi_n(x)\}$中找一函数$S^*(x)$，使误差平方和

$$\begin{aligned}\|\boldsymbol{\delta}\|_2^2 &= \sum_{i=0}^{m}\delta_i^2 = \sum_{i=0}^{m}[S^*(x_i)-y_i]^2 \\ &= \min_{S(x)\in\varphi}\sum_{i=0}^{m}[S(x_i)-y_i]^2, \end{aligned} \tag{5.1}$$

这里

$$S(x) = a_0\varphi_0(x)+a_1\varphi_1(x)+\cdots+a_n\varphi_n(x) \quad (n<m). \tag{5.2}$$

这就是一般的最小二乘逼近，用几何语言说，就称为曲线拟合的最小二乘法.

用最小二乘法求拟合曲线时，首先要确定$S(x)$的形式. 这不单纯是数学问题，还与所研究问题的运动规律及所得观测数据(x_i, y_i)有关；通常要从问题的运动规律及给定数据描图，确定$S(x)$的形式，并通过实际计算选出较好的结果——这点将从下面的例题得到说明. $S(x)$的一般表达式为(5.2)表示的线性形式. 若$\varphi_k(x)$是k次多项式，$S(x)$就是n次多项式. 为了使问题的提法更有一般性，通常在最小二乘法中$\|\delta\|_2^2$都考虑为加权平方和

$$\|\boldsymbol{\delta}\|_2^2 = \sum_{i=0}^{m}\omega(x_i)[S(x_i)-f(x_i)]^2. \tag{5.3}$$

这里$\omega(x)\geqslant 0$是$[a,b]$上的权函数，它表示不同点$(x_i, f(x_i))$处的数据比重不同，例如，$\omega(x_i)$可表示在点$(x_i, f(x_i))$处重复观测的次数. 用最小二乘法求拟合曲线的问题，就是在形如(5.2)的

$S(x)$中求一函数 $y=S^*(x)$,使(5.3)取得最小. 它转化为求多元函数

$$I(a_0,a_1,\cdots,a_n) = \sum_{i=0}^{m}\omega(x_i)\left[\sum_{j=0}^{n}a_j\varphi_j(x_i)-f(x_i)\right]^2 \tag{5.4}$$

的极小点$(a_0^*,a_1^*,\cdots,a_n^*)$问题. 这与第 4 节讨论的问题完全类似. 由求多元函数极值的必要条件,有

$$\frac{\partial I}{\partial a_k} = 2\sum_{i=0}^{m}\omega(x_i)\left[\sum_{j=0}^{n}a_j\varphi_j(x_i)-f(x_i)\right]\varphi_k(x_i) = 0$$

$$(k=0,1,\cdots,n).$$

若记
$$(\varphi_j,\varphi_k) = \sum_{i=0}^{m}\omega(x_i)\varphi_j(x_i)\varphi_k(x_i), \tag{5.5}$$

$$(f,\varphi_k) = \sum_{i=0}^{m}\omega(x_i)f(x_i)\varphi_k(x_i) \equiv d_k$$

$$(k=0,1,\cdots,n).$$

上式可改写为

$$\sum_{j=0}^{n}(\varphi_k,\varphi_j)a_j = d_k \qquad (k=0,1,\cdots,n). \tag{5.6}$$

这方程称为**法方程**,可写成矩阵形式

$$\boldsymbol{G}\boldsymbol{a} = \boldsymbol{d}.$$

其中 $\boldsymbol{a}=(a_0,a_1,\cdots,a_n)^{\mathrm{T}}$,$\boldsymbol{d}=(d_0,d_1,\cdots,d_n)^{\mathrm{T}}$,

$$\boldsymbol{G} = \begin{bmatrix} (\varphi_0,\varphi_0) & (\varphi_0,\varphi_1) & \cdots & (\varphi_0,\varphi_n) \\ (\varphi_1,\varphi_0) & (\varphi_1,\varphi_1) & \cdots & (\varphi_1,\varphi_n) \\ \vdots & \vdots & & \vdots \\ (\varphi_n,\varphi_0) & (\varphi_n,\varphi_1) & \cdots & (\varphi_n,\varphi_n) \end{bmatrix}. \tag{5.7}$$

要使法方程(5.6)有唯一解 $a_0,a_1,\cdots,a_n$,就要求矩阵 $\boldsymbol{G}$ 非奇异,必须指出,$\varphi_0(x),\varphi_1(x),\cdots,\varphi_n(x)$在$[a,b]$上线性无关不能推出矩阵 $\boldsymbol{G}$ 非奇异. 例如,令 $\varphi_0(x)=\sin x$, $\varphi_1(x)=\sin 2x$, $x\in$

$[0,2\pi]$，显然$\{\varphi_0(x),\varphi_1(x)\}$在$[0,2\pi]$上线性无关，但若取点 $x_k=k\pi,k=0,1,2(n=1,m=2)$，那么有 $\varphi_0(x_k)=\varphi_1(x_k)=0,k=0,1,2$，由此得出

$$G=\begin{vmatrix}(\varphi_0,\varphi_0) & (\varphi_0,\varphi_1)\\(\varphi_1,\varphi_0) & (\varphi_1,\varphi_1)\end{vmatrix}=0.$$

为保证(5.6)的系数矩阵 G 非奇异，必须加上另外的条件.

定义 10　设 $\varphi_0(x),\varphi_1(x),\cdots,\varphi_n(x)\in C[a,b]$的任意线性组合在点集$\{x_i,i=0,1,\cdots,m\}(m\geqslant n)$上至多只有 n 个不同的零点，则称 $\varphi_0(x),\varphi_1(x),\cdots,\varphi_n(x)$在点集$\{x_i,i=0,1,\cdots,m\}$上满足**哈尔(Haar)条件.**

显然 $1,x,\cdots,x^n$ 在任意 $m(m\geqslant n)$个点上满足哈尔条件.

可以证明，如果 $\varphi_0(x),\varphi_1(x),\cdots,\varphi_n(x)\in C[a,b]$在$\{x_i\}_0^m$ 上满足 Haar 条件，则法方程(5.6)的系数矩阵(5.7)非奇异，于是方程(5.6)存在唯一的解 $a_k=a_k^*,k=0,1,\cdots,n$. 从而得到函数 $f(x)$的最小二乘解为

$$S^*(x)=a_0^*\varphi_0(x)+a_1^*\varphi_1(x)+\cdots+a_n^*\varphi_n(x).$$

可以证明这样得到的 $S^*(x)$，对任何形如(5.2)的 $S(x)$，都有

$$\sum_{i=0}^m\omega(x_i)[S^*(x_i)-f(x_i)]^2\leqslant\sum_{i=0}^m\omega(x_i)[S(x_i)-f(x_i)]^2,$$

故 $S^*(x)$确是所求最小二乘解. 它的证明与(4.4)式相似，读者可自己完成.

给定 $f(x)$的离散数据$\{(x_i,y_i),i=0,1,\cdots,m\}$，要确定 φ 是困难的，一般可取 $\varphi=\mathrm{span}\{1,x,\cdots,x^n\}$，但这样做当 $n\geqslant3$ 时，求解法方程(5.6)与连续情形一样，将出现系数矩阵 G 为病态的问题，通常对 $n=1$ 的简单情形都可通过求法方程(5.6)得到$S^*(x)$. 有时根据给定数据图形，其拟合函数 $y=S(x)$表面上不是(5.2)的形式，但通过变换仍可化为线性模型. 例如，$S(x)=ae^{bx}$，若两边取对数得

$$\ln S(x) = \ln a + bx,$$

它就是形如(5.2)的线性模型,具体做法见例 8.

例 7 已知一组实验数据如下,求它的拟合曲线.

x_i	1	2	3	4	5
f_i	4	4.5	6	8	8.5
φ_i	2	1	3	1	1

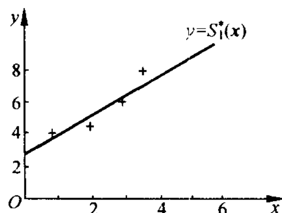


图 3-4

解 根据所给数据,在坐标纸上标出,见图3-4. 从图中看到各点在一条直线附近,故可选择线性函数作拟合曲线,即令 $S_1(x)=a_0+a_1x$,这里 $m=4,n=1,\varphi_0(x)=1,\varphi_1(x)=x$,故

$$(\varphi_0,\varphi_0)=\sum_{i=0}^{4}\omega_i=8,(\varphi_0,\varphi_1)=(\varphi_1,\varphi_0)=\sum_{i=0}^{4}\omega_i x_i=22,$$

$$(\varphi_1,\varphi_1)=\sum_{i=0}^{4}\omega_i x_i^2=74,(\varphi_0,f)=\sum_{i=0}^{4}\omega_i f_i=47,$$

$$(\varphi_1,f)=\sum_{i=0}^{4}\omega_i x_i f_i=145.5.$$

由(5.6)得方程组

$$\begin{cases}8a_0+22a_1=47,\\22a_0+74a_1=145.5.\end{cases}$$

解得 $a_0=2.77,a_1=1.13$. 于是所求拟合曲线为

$$S_1^*(x)=2.77+1.13x.$$

例 8 设数据$(x_i,y_i)(i=0,1,2,3,4)$由表 3-1 给出,表中第 4 行为 $\ln y_i=\bar{y}_i$,可以看出数学模型为 $y=ae^{bx}$,用最小二乘法确定 a 及 b.

解 根据给定数据$(x_i,y_i)(i=0,1,2,3,4)$描图可确定拟合曲线方程为 $y=ae^{bx}$,它不是线性形式. 两边取对数得 $\ln y=\ln a+bx$,若令 $\bar{y}=\ln y,A=\ln a$,则得 $\bar{y}=A+bx,\varphi=\{1,x\}$. 为确定 A,b,先将(x_i,y_i)转化为$(x_i,\bar{y}_i)$,数据表见表 3-1.

表 3-1

i	0	1	2	3	4
x_i	1.00	1.25	1.50	1.75	2.00
y_i	5.10	5.79	6.53	7.45	8.46
$\bar{y}_i$	1.629	1.756	1.876	2.008	2.135

根据最小二乘法,取 $\varphi_0(x)=1,\varphi_1(x)=x,\omega(x)\equiv 1$,得

$$(\varphi_0,\varphi_0)=5,$$

$$(\varphi_0,\varphi_1)=\sum_{i=0}^{4}x_i=7.5,$$

$$(\varphi_1,\varphi_1)=\sum_{i=0}^{4}x_i^2=11.875,$$

$$(\varphi_0,\bar{y})=\sum_{i=0}^{4}\bar{y}_i=9.404,$$

$$(\varphi_1,\bar{y})=\sum_{i=0}^{4}x_i\bar{y}_i=14.422.$$

故有法方程

$$\begin{cases}5A+7.50b=9.404,\\7.50A+11.875b=14.422.\end{cases}$$

解得 $A=1.122,b=0.505,a=e^A=3.071$. 于是得最小二乘拟合曲线为

$$y = 3.071e^{0.505x}.$$

现在很多计算机配有自动选择数学模型的程序，其方法与本例同. 程序中因变量与自变量变换的函数类型较多，通过计算比较误差找到拟合得较好的曲线，最后输出曲线图形及数学表达式.

3.5.2 用正交多项式做最小二乘拟合

用最小二乘法得到的法方程组(5.6)，其系数矩阵 $\boldsymbol{G}$ 是病态的，但如果 $\varphi_0(x), \varphi_1(x), \cdots, \varphi_n(x)$ 是关于点集 $\{x_i\}(i=0,1,\cdots,m)$ 带权 $\omega(x_i)(i=0,1,\cdots,m)$ 正交的函数族，即

$$(\varphi_j, \varphi_k) = \sum_{i=0}^{m} \omega(x_i)\varphi_j(x_i)\varphi_k(x_i) = \begin{cases} 0, & j \neq k, \\ A_k > 0, & j = k, \end{cases} \tag{5.8}$$

则方程(5.6)的解为

$$a_k^* = \frac{(f, \varphi_k)}{(\varphi_k, \varphi_k)} = \frac{\sum_{i=0}^{m} \omega(x_i) f(x_i) \varphi_k(x_i)}{\sum_{i=0}^{m} \omega(x_i) \varphi_k^2(x_i)} \quad (k = 0, 1, \cdots, n), \tag{5.9}$$

且平方误差为

$$\| \boldsymbol{\delta} \|_2^2 = \| f \|_2^2 - \sum_{k=0}^{n} A_k (a_k^*)^2.$$

现在我们根据给定节点 $x_0, x_1, \cdots, x_m$ 及权函数 $\omega(x) > 0$，造出带权 $\omega(x)$ 正交的多项式 $\{P_n(x)\}$. 注意 $n \leqslant m$，用递推公式表示 $P_k(x)$，即

$$\begin{cases} P_0(x) = 1, \\ P_1(x) = (x - \alpha_1) P_0(x), \\ P_{k+1}(x) = (x - \alpha_{k+1}) P_k(x) - \beta_k P_{k-1}(x) \\ \qquad (k = 1, 2, \cdots, n-1). \end{cases} \tag{5.10}$$

这里 $P_k(x)$ 是首项系数为 1 的 k 次多项式，根据 $P_k(x)$ 的正交

性，得

$$\begin{cases} \alpha_{k+1} = \dfrac{\sum\limits_{i=0}^{m}\omega(x_i)x_iP_k^2(x_i)}{\sum\limits_{i=0}^{m}\omega(x_i)P_k^2(x_i)} = \dfrac{(xP_k(x),P_k(x))}{(P_k(x),P_k(x))} \\ \quad = \dfrac{(xP_k,P_k)}{(P_k,P_k)} \quad (k=0,1,\cdots,n-1), \\ \beta_k = \dfrac{\sum\limits_{i=0}^{m}\omega(x_i)P_k^2(x_i)}{\sum\limits_{i=0}^{m}\omega(x_i)P_{k-1}^2(x_i)} = \dfrac{(P_k,P_k)}{(P_{k-1},P_{k-1})} \\ \quad (k=1,\cdots,n-1). \end{cases} \tag{5.11}$$

下面用归纳法证明这样给出的$\{P_k(x)\}$是正交的. 由(5.10)第二式及(5.11)中α_1的表达式，有

$$\begin{aligned}(P_0,P_1) &= (P_0,xP_0) - \alpha_1(P_0,P_0) \\ &= (P_0,xP_0) - \frac{(xP_0,P_0)}{(P_0,P_0)}(P_0,P_0) = 0.\end{aligned}$$

现假定$(P_l,P_s)=0(l\neq s)$对$s=0,1,\cdots,l-1$及$l=0,1,\cdots,k$; $k<n$均成立，要证$(P_{k+1},P_s)=0$对$s=0,1,\cdots,k$均成立. 由(5.10)有

$$\begin{aligned}(P_{k+1},P_s) &= ((x-\alpha_{k+1})P_k,P_s) - \beta_k(P_{k-1},P_s) \\ &= (xP_k,P_s) - \alpha_{k+1}(P_k,P_s) - \beta_k(P_{k-1},P_s).\end{aligned} \tag{5.12}$$

由归纳法假定，当$0\leqslant s\leqslant k-2$时

$$(P_k,P_s) = 0, \quad (P_{k-1},P_s) = 0.$$

另外，$xP_s(x)$是首项系数为1的$s+1$次多项式，它可由P_0, $P_1,\cdots,P_{s+1}$的线性组合表示，而$s+1\leqslant k-1$，故由归纳法假定又有

$$(xP_k,P_s) \equiv (P_k,xP_s) = 0,$$

于是由(5.12)，当$s\leqslant k-2$时，$(P_{k+1},P_s)=0$.

再看

$$(P_{k+1},P_{k-1})=(xP_k,P_{k-1})-\alpha_{k+1}(P_k,P_{k-1}) \\ -\beta_k(P_{k-1},P_{k-1}), \tag{5.13}$$

由假定有 $$(P_k,P_{k-1})=0,$$

$$(xP_k,P_{k-1})=(P_k,xP_{k-1})=\Big(P_k,P_k+\sum_{j=0}^{k-1}c_jP_j\Big)=(P_k,P_k).$$

利用(5.11)中 β_k 表达式及以上结果,得

$$\begin{aligned}(P_{k+1},P_{k-1})&=(xP_k,P_{k-1})-\beta_k(P_{k-1},P_{k-1})\\&=(P_k,P_k)-(P_k,P_k)=0.\end{aligned}$$

最后,由(5.11)有

$$\begin{aligned}(P_{k+1},P_k)&=(xP_k,P_k)-\alpha_{k+1}(P_k,P_k)-\beta_k(P_k,P_{k-1})\\&=(xP_k,P_k)-\frac{(xP_k,P_k)}{(P_k,P_k)}(P_k,P_k)=0.\end{aligned}$$

至此已证明了由(5.10)及(5.11)确定的多项式$\{P_k(x)\}(k=0,1,\cdots,n,n\leqslant m)$组成一个关于点集$\{x_i\}$的正交系.

用正交多项式$\{P_k(x)\}$的线性组合作最小二乘曲线拟合,只要根据公式(5.10)及(5.11)逐步求 $P_k(x)$的同时,相应计算出系数

$$a_k=\frac{(f,P_k)}{(P_k,P_k)}=\frac{\sum_{i=0}^{m}\omega(x_i)f(x_i)P_k(x_i)}{\sum_{i=0}^{m}\omega(x_i)P_k^2(x_i)}\quad(k=0,1,\cdots,n),$$

并逐步把 $a_k^* P_k(x)$累加到 $S(x)$中去,最后就可得到所求的拟合曲线

$$y=S(x)=a_0^*P_0(x)+a_1^*P_1(x)+\cdots+a_n^*P_n(x).$$

这里 n 可事先给定或在计算过程中根据误差确定. 用这种方法编程序不用解方程组,只用递推公式,并且当逼近次数增加一次时,只要把程序中循环数加 1,其余不用改变. 这是目前用多项式做曲线拟合最好的计算方法,有通用的语言程序供用户使用.

3.6 最佳平方三角逼近与快速傅里叶变换

当 $f(x)$ 是周期函数时，显然用三角多项式逼近 $f(x)$ 比用代数多项式更合适，本节主要讨论用三角多项式做最小平方逼近及**快速傅里叶变换**(Fast Fourier Transform)简称 FFT 算法.

3.6.1 最佳平方三角逼近与三角插值

设 $f(x)$ 是以 2π 为周期的平方可积函数，用三角多项式

$$S_n(x) = \frac{1}{2}a_0 + a_1\cos x + b_1\sin x + \cdots + a_n\cos nx + b_n\sin nx \tag{6.1}$$

做最佳平方逼近函数. 由于三角函数族

$$1,\cos x,\sin x,\cdots,\cos kx,\sin kx,\cdots$$

在$[0,2\pi]$上是正交函数族，于是 $f(x)$ 在$[0,2\pi]$上的最小平方三角逼近多项式 $S_n(x)$ 的系数是

$$\begin{aligned} a_k &= \frac{1}{\pi}\int_0^{2\pi} f(x)\cos kx\,\mathrm{d}x \qquad (k = 0,1,\cdots,n), \\ b_k &= \frac{1}{\pi}\int_0^{2\pi} f(x)\sin kx\,\mathrm{d}x \qquad (k = 1,\cdots,n), \end{aligned} \tag{6.2}$$

a_k,b_k 称为傅里叶系数，函数 $f(x)$ 按傅里叶系数展开得到的级数

$$\frac{1}{2}a_0 + \sum_{k=1}^{\infty}(a_k\cos kx + b_k\sin kx) \tag{6.3}$$

就称为傅里叶级数，只要 $f'(x)$ 在$[0,2\pi]$上分段连续，则级数(6.3)一致收敛到 $f(x)$.

对于最佳平方逼近多项式(6.1)有

$$\| f(x) - S_n(x) \|_2^2 = \| f(x) \|_2^2 - \| S_n(x) \|_2^2.$$

由此可以得到相应于(4.11)的贝塞尔不等式

$$\frac{1}{2}a_0^2 + \sum_{k=1}^{n}(a_k^2 + b_k^2) \leqslant \frac{1}{\pi}\int_0^{2\pi}[f(x)]^2\mathrm{d}x.$$

因为右边不依赖于 n，左边单调有界，所以级数 $\frac{1}{2}a_0^2 + \sum_{k=1}^{\infty}(a_k^2 + b_k^2)$ 收敛，并有

$$\lim_{k\to\infty}a_k = \lim_{k\to\infty}b_k = 0.$$

当 $f(x)$ 只在给定的离散点集 $\left\{x_j = \frac{2\pi}{N}j, j=0,1,\cdots,N-1\right\}$ 上已知时，则可类似得到离散点集正交性与相应的离散傅里叶系数. 为方便起见，下面只给出奇数个点的情形. 令

$$x_j = \frac{2\pi j}{2m+1} \quad (j = 0,1,\cdots,2m),$$

可以证明对任何 $0 \leqslant k, l \leqslant m$ 成立

$$\begin{cases} \sum_{j=0}^{2m}\sin lx_j \sin kx_j = \begin{cases} 0, & l \neq k, l = k = 0; \\ \frac{2m+1}{2}, & l = k \neq 0; \end{cases} \\ \sum_{j=0}^{2m}\cos lx_j \cos kx_j = \begin{cases} 0, & l \neq k; \\ \frac{2m+1}{2}, & l = k \neq 0; \\ 2m+1, & l = k = 0; \end{cases} \\ \sum_{j=0}^{2m}\cos lx_j \sin kx_j = 0, \quad 0 \leqslant k, j \leqslant m. \end{cases}$$

这就表明函数族 $\{1, \cos x, \sin x, \cdots, \cos mx, \sin mx\}$ 在点集 $\left\{x_j = \frac{2\pi j}{2m+1}\right\}$ 上正交，若令 $f_j = f(x_j) \quad (j = 0, 1, \cdots, 2m)$，则 $f(x)$ 的最小二乘三角逼近为

$$S_n(x) = \frac{1}{2}a_0 + \sum_{k=1}^{n}(a_k\cos kx + b_k\sin kx), \quad n < m,$$

其中

$$a_k = \frac{2}{2m+1}\sum_{j=0}^{2m} f_j \cos\frac{2\pi jk}{2m+1} \quad (k=0,1,\cdots,m),$$

$$b_k = \frac{2}{2m+1}\sum_{j=0}^{2m} f_j \sin\frac{2\pi jk}{2m+1} \quad (k=1,\cdots,n). \tag{6.4}$$

当 $n=m$ 时，可证明

$$S_m(x_j) = f_j \qquad (j=0,1,\cdots,2m),$$

于是

$$S_m(x) = \frac{1}{2}a_0 + \sum_{k=1}^{m}(a_k \cos kx + b_k \sin kx)$$

就是三角插值多项式，系数仍由(6.4)表示.

更一般情形，假定 $f(x)$是以 2π 为周期的复函数，给定在 N 个等分点 $x_j=\frac{2\pi}{N}j\,(j=0,1,\cdots,N-1)$上的值 $f_j=f\left(\frac{2\pi}{N}j\right)$，由于

$$e^{ijx} = \cos(jx) + i\sin(jx) \qquad (j=0,1,\cdots,N-1, i=\sqrt{-1}),$$

函数族$\{1,e^{ix},\cdots,e^{i(N-1)x}\}$在区间$[0,2\pi]$上是正交的，函数 e^{ijx} 在等距点集 $x_k=\frac{2\pi}{N}k\,(k=0,1,\cdots,N-1)$上的值 e^{ijx_k} 组成的向量记作

$$\boldsymbol{\Phi}_j = (1, e^{ij\frac{2\pi}{N}}, \cdots, e^{ij\frac{2\pi}{N}(N-1)})^{\mathrm{T}}.$$

当 $j=0,1,\cdots,N-1$ 时，N 个复向量$\boldsymbol{\Phi}_0,\boldsymbol{\Phi}_1,\cdots,\boldsymbol{\Phi}_{N-1}$具有下面所定义的正交性：

$$(\boldsymbol{\Phi}_l,\boldsymbol{\Phi}_s) = \sum_{k=0}^{N-1} e^{il\frac{2\pi}{N}k} e^{-is\frac{2\pi}{N}k} = \sum_{k=0}^{N-1} e^{i(l-s)\frac{2\pi}{N}k}$$

$$= \begin{cases} 0, l \neq s; \\ N, l = s. \end{cases} \tag{6.5}$$

事实上，令 $r=e^{i(l-s)\frac{2\pi}{N}}$，若 $0\leqslant l,s\leqslant N-1$，则有

$$0 \leqslant l \leqslant N-1, \qquad -(N-1) \leqslant -s \leqslant 0,$$

于是

$$-(N-1) \leqslant l-s \leqslant N-1,$$

即

$$-1 < -\frac{N-1}{N} \leqslant \frac{l-s}{N} \leqslant \frac{N-1}{N} < 1;$$

若 $l-s\neq 0$,则 $r\neq 1$,从而

$$r^N = e^{i(l-s)2\pi} = 1;$$

于是

$$(\boldsymbol{\phi}_l,\boldsymbol{\phi}_s) = \sum_{k=0}^{N-1} r^k = \frac{1-r^N}{1-r} = 0.$$

若 $l=s$,则 $r=1$,于是

$$(\boldsymbol{\phi}_s,\boldsymbol{\phi}_s) = \sum_{k=0}^{N-1} r^k = N.$$

这就证明了(6.5)成立. 即 $\boldsymbol{\phi}_0,\boldsymbol{\phi}_1,\cdots,\boldsymbol{\phi}_{N-1}$ 是正交的.

因此,$f(x)$ 在 N 个点 $\left\{x_j=\frac{2\pi}{N}j,j=0,1,\cdots,N-1\right\}$ 上的最小二乘傅里叶逼近为

$$S(x) = \sum_{k=0}^{n-1} c_k e^{ikx}, \quad n\leqslant N, \tag{6.6}$$

其中

$$c_k = \frac{1}{N}\sum_{j=0}^{N-1} f_j e^{-ikj\frac{2\pi}{N}} \quad (k=0,1,\cdots,n-1). \tag{6.7}$$

在(6.6)中若 $n=N$,则 $S(x)$ 为 $f(x)$ 在点 $x_j(j=0,1,\cdots,N-1)$ 上的插值函数,即 $S(x_j)=f(x_j)$,于是由(6.6)得

$$f_j = \sum_{k=0}^{N-1} c_k e^{ik\frac{2\pi}{N}j} \qquad (j=0,1,\cdots,N-1). \tag{6.8}$$

(6.7)是由 $\{f_j\}$ 求 $\{c_k\}$ 的过程,称为 $f(x)$ 的**离散傅里叶变换**. 简称 DFT,而(6.8)是由 $\{c_k\}$ 求 $\{f_j\}$ 的过程,称为**反变换**. 它们是使用计算机进行傅里叶分析(简称傅氏分析)的主要方法,在数字讯号处理,全息技术,光谱和声谱分析等很多领域都有广泛应用.

3.6.2 快速傅氏变换(FFT)

不论是按(6.7)式由 $\{f_k\}$ 求 $\{c_j\}$ 或是按(6.8)式由 $\{c_j\}$ 求 $\{f_k\}$,还是由(6.4)计算傅里叶逼近系数 a_k,b_k 都可归结为计算

$$c_j = \sum_{k=0}^{N-1} x_k w^{kj} \qquad (j = 0,1,\cdots,N-1), \qquad (6.9)$$

其中 $w=\exp(-\mathrm{i}2\pi/N)$(正变换)或 $w=\exp(\mathrm{i}2\pi/N)$(反变换),$\{x_k\}(k=0,1,\cdots,N-1)$是已知复数序列. 如直接用(6.9)计算 c_j,需要 N 次复数乘法和 N 次复数加法,称为 N 个操作,计算全部 c_j 共要 N^2 个操作. 当 N 较大且处理数据很多时,就是用高速的电子计算机,很多实际问题仍然无法计算,直到 20 世纪 60 年代中期产生了 FFT 算法,大大提高了运算速度,才使傅氏变换得以广泛应用. FFT 算法的思想就是尽量减少乘法次数,例如计算 $ab+ac=a(b+c)$,用左端计算时做两次乘法,右端只用一次乘法. 用公式(6.9)计算全部 c_j,表面看要做 N^2 个乘法,实际上所有 $\exp(\mathrm{i}2\pi kj/N)$, $j,k=0,1,\cdots,N-1$ 中,只有 N 个不同的值 w^0, $w^1,\cdots,w^{N-1}$,特别当 $N=2^p$ 时,只有 $N/2$ 个不同的值. 因此,我们可把同一个 w^r 对应的 x_k 相加后再乘 w^r 这就能大量减少乘法次数. 为了具体推导 FFT 算法,先给出定义.

设正整数 m 除以 N 后得商 q 及余数 r,则 $m=qN+r$,r 称为 m 的 N 同余数,以 $m\overset{N}{=}r$ 表示. 由于 $w=\exp(\mathrm{i}2\pi/N)$,$w^N=\mathrm{e}^{\mathrm{i}2\pi}=1$,故有 $w^m=(w^N)^q w^r=w^r$.

因此计算 w^m 时可用 w 的 N 同余数 r 代替 m,从而推出 FFT 算法. 下面以 $N=2^3$ 为例. 说明 FFT 的计算方法. 由于 $0\leqslant k,j\leqslant N-1=2^3-1=7$,则(6.9)的和是

$$c_j = \sum_{k=0}^{7} x_k w^{jk} \quad (j = 0,1,\cdots,7). \qquad (6.10)$$

将 k,j 用二进制表示为

$$k = k_2 2^2 + k_1 2^1 + k_0 2^0 = (k_2 k_1 k_0),$$

$$j = j_2 2^2 + j_1 2^1 + j_0 2^0 = (j_2 j_1 j_0);$$

其中 $k_r, j_r (r=0,1,2)$ 只能取 0 或 1,例如 $6=2^2+2^1+0\cdot 2^0=(110)$. 根据 k,j 表示法,有

$$c_j = c(j_2 j_1 j_0), x_k = x(k_2 k_1 k_0).$$

公式(6.10)可表示为

$$\begin{aligned} & c(j_2 j_1 j_0) \\ &= \sum_{k_0=0}^{1}\sum_{k_1=0}^{1}\sum_{k_2=0}^{1} x(k_2 k_1 k_0) w^{(k_2 k_1 k_0)(j_2 2^2 + j_1 2^1 + j_0 2^0)} \\ &= \sum_{k_0=0}^{1}\left\{\sum_{k_1=0}^{1}\left[\sum_{k_2=0}^{1} x(k_2 k_1 k_0) w^{j_0(k_2 k_1 k_0)}\right] w^{j_1(k_1 k_0 0)}\right\} \\ &\quad \cdot w^{j_2(k_0 00)}. \end{aligned} \tag{6.11}$$

若引入记号

$$\begin{cases} A_0(k_2 k_1 k_0) = x(k_2 k_1 k_0), \\ A_1(k_1 k_0 j_0) = \sum\limits_{k_2=0}^{1} A_0(k_2 k_1 k_0) w^{j_0(k_2 k_1 k_0)}, \\ A_2(k_0 j_1 j_0) = \sum\limits_{k_1=0}^{1} A_1(k_1 k_0 j_0) w^{j_1(k_1 k_0 0)}, \\ A_3(j_2 j_1 j_0) = \sum\limits_{k_0=0}^{1} A_2(k_0 j_1 j_0) w^{j_2(k_0 00)}. \end{cases} \tag{6.12}$$

则(6.11)变成

$$c(j_2 j_1 j_0) = A_3(j_2 j_1 j_0).$$

它说明利用 N 同余数可把计算 c_j 分为 p 步，用公式(6.12)计算，每计算一个 A_q 只用 2 次复数乘法，计算一个 c_j 用 $2p$ 次复数乘法，计算全部 c_j 共用 $2pN$ 次复数乘法. 若注意 $w^{j_0 2^{p-1}} = w^{j_0 N/2} = (-1)^j$，公式(6.12)还可进一步简化为

$$\begin{aligned} A_1(k_1 k_0 j_0) &= \sum_{k_2=0}^{1} A_0(k_2 k_1 k_0) w^{j_0(k_2 k_1 k_0)} \\ &= A_0(0 k_1 k_0) w^{j_0(0 k_1 k_0)} \\ &\quad + A_0(1 k_1 k_0) w^{j_0 2^2} w^{j_0(0 k_1 k_0)} \\ &= [A_0(0 k_1 k_0) + (-1)^{j_0} A_0(1 k_1 k_0)] w^{j_0(0 k_1 k_0)}, \end{aligned}$$

$$A_1(k_1k_00)=A_0(0k_1k_0)+A_0(1k_1k_0),$$
$$A_1(k_1k_01)=[A_0(0k_1k_0)-A_0(1k_1k_0)]w^{(0k_1k_0)}.$$

将这表达式中二进制表示还原为十进制表示：$k=(0k_1k_0)=k_12^1+k_02^0$，即 $k=0,1,2,3$，得

$$\begin{cases}A_1(2k)=A_0(k)+A_0(k+2^2),\\ A_1(2k+1)=[A_0(k)-A_0(k+2^2)]w^k\\ \qquad\qquad (k=0,1,2,3).\end{cases}\tag{6.13}$$

同样(6.12)中的 A_2 也可简化为

$$A_2(k_0j_1j_0)=[A_1(0k_0j_0)+(-1)^{j_1}A_1(1k_0j_0)]w^{j_1(0k_00)},$$

即

$$A_2(k_00j_0)=A_1(0k_0j_0)+A_1(1k_0j_0),$$
$$A_2(k_01j_0)=[A_1(0k_0j_0)-A_1(1k_0j_0)]w^{(0k_00)}.$$

把二进制表示还原为十进制表示，得

$$\begin{cases}A_2(k2^2+j)=A_1(2k+j)+A_1(2k+j+2^2),\\ A_2(k2^2+j+2)=[A_1(2k+j)-A_1(2k+j+2^2)]w^{2k}\end{cases}$$
$$(k=0,1;j=0,1).\tag{6.14}$$

同理(6.12)中 A_3 可简化为

$$A_3(j_2j_1j_0)=A_2(0j_1j_0)+(-1)^{j_2}A_2(1j_1j_0),$$

即

$$A_3(0j_1j_0)=A_2(0j_1j_0)+A_2(1j_1j_0),$$
$$A_3(1j_1j_0)=A_2(0j_1j_0)-A_2(1j_1j_0).$$

表示为十进制，有

$$\begin{cases}A_3(j)=A_2(j)+A_2(j+2^2),\\ A_3(j+2^2)=A_2(j)-A_2(j+2^2)\end{cases}$$
$$(j=0,1,2,3).\tag{6.15}$$

根据公式(6.13)，(6.14)，(6.15)，由 $A_0(k)=x(k)=x_k(k=0,1,\cdots,7)$ 逐次计算到 $A_3(j)=c_j(j=0,1,\cdots,7)$，见表 3-2.

上面推导的 $N=2^3$ 的计算公式可类似地推广到 $N=2^p$ 的情形. 根据公式(6.13),(6.14),(6.15),一般情况的FFT计算公式如下:

$$\begin{cases} A_q(k2^q+j)=A_{q-1}(k2^{q-1}+j)+A_{q-1}(k2^{q-1}+j+2^{p-1}), \\ A_q(k2^q+j+2^{q-1}) \\ =[A_{q-1}(k2^{q-1}+j)-A_{q-1}(k2^{q-1}+j+2^{p-1})]w^{k2^{q-1}}, \end{cases} \tag{6.16}$$

其中 $q=1,\cdots,p;k=0,1,\cdots,2^{p-q}-1;j=0,1,\cdots,2^{q-1}-1$. A_q 括号内的数代表它的位置,在计算机中代表存放数的地址. 一组 A_q 占用 N 个复数单元,计算时需给出两组单元,从 $A_0(m)(m=0,1,\cdots,N-1)$出发,q 由 1 到 p 算到 $A_p(j)=c_j(j=0,1,\cdots,N-1)$,即为所求. 计算过程中只要按地址号存放 A_q,则最后得到的 $A_p(j)$就是所求离散频谱的次序.(注意,目前一些计算机程序计算结果地址是逆序排列,还要增加倒地址的一步才是我们这里介绍的结果). 这个计算公式除了具有不倒地址的优点外,计算只有两重循环,外循环 q 由 1 计算到 p,内循环 k 由 0 计算到 $2^{p-q}-1$,j 由 0 计算到 $2^{q-1}-1$,更重要的是整个计算过程省计算量. 由公式看到算一个 A_q 共做 $2^{p-q}2^{q/1}=N/2$ 次复数乘法,而最后一步计算 A_p 时,由于 $w^{k2^{p-1}}=(w^{N/2})^k=(-1)^k=(-1)^0=1$(注意 $q=p$ 时 $2^{p-q}-1=0$,故 $k=0$),因此,总共要算$(p-1)N/2$ 次复数乘法,它比直接用(6.9)需 N^2 次乘法快得多,计算量比值是 $N:(p-1)/2$. 当 $N=2^{10}$时比值是 $1024:4.5\approx230:1$,它比一般 FFT 的计算量(pN 次乘法)也快一倍. 我们称(6.16)的计算公式为**改进的 FFT 算法**,下面给出这一算法的程序步骤:

步骤 1 给出数组 $A_1(N)$,$A_2(N)$及 $w(N/2)$;

步骤 2 将已知的记录复数数组$\{x_k\}$输入到单元 $A_1(k)$中(k 从 0 到 $N-1$);

表 3-2

单元码号	0 000	1 001	2 010	3 011	4 100	5 101	6 110	7 111
					$w^0=1$	w^1	w^2	w^3
$x_k=A_0(k)$	$A_0(0)$	$A_0(1)$	$A_0(2)$	$A_0(3)$	$A_0(4)$	$A_0(5)$	$A_0(6)$	$A_0(7)$
A_1	$A_0(0)+$ $A_0(4)$	$[A_0(0)-$ $A_0(4)]w^0$	$A_0(1)+$ $A_0(5)$	$[A_0(1)-$ $A_0(5)]w^1$	$A_0(2)+$ $A_0(6)$	$[A_0(2)-$ $A_0(6)]w^2$	$A_0(3)+$ $A_0(7)$	$[A_0(3)-$ $A_0(7)]w^3$
A_2	$A_1(0)+$ $A_1(4)$	$A_1(1)+$ $A_1(5)$	$[A_1(0)-$ $A_1(4)]w^0$	$[A_1(1)-$ $A_1(5)]w^0$	$A_1(2)+$ $A_1(6)$	$A_1(3)+$ $A_1(7)$	$[A_1(2)-$ $A_1(6)]w^2$	$[A_1(3)-$ $A_1(7)]w^2$
$c_i=A_3(j)$	$(0)+(4)$	$(1)+(5)$	$(2)+(6)$	$(3)+(7)$	$(0)-(4)$	$(1)-(5)$	$(2)-(6)$	$(3)-(7)$

步骤 3 计算 $w^m=\exp\left(-\mathrm{i}\,\frac{2\pi}{N}m\right)\left(\text{或 } w^m=\exp\left(\mathrm{i}\,\frac{2\pi}{N}m\right)\right)$存放在单元 $w(m)$ 中(m 从 0 到 $(N/2)-1$)；

步骤 4 q 循环从 1 到 p，若 q 为奇数做步骤 5，否则做步骤 6；

步骤 5 k 循环从 0 到 $2^{p-q}-1$，j 循环从 0 到 $2^{q-1}-1$，计算

$$A_2(k2^q+j) = A_1(k2^{q-1}-j)+A_1(k2^{q-1}+j+2^{p-1}),$$

$$A_2(k2^q+j+2^{q-1}) = [A_1(k2^{q-1}+j) - A_1(k2^{q-1}+j+2^{q-1})]w(k2^{q-1});$$

转步骤 7.

步骤 6 k 循环从 0 到 $2^{p-q}-1$，j 循环从 0 到 2^{q-1}，计算

$$A_1(k2^q+j) = A_2(k2^{q-1}+j)+A_2(k2^{q-1}+j+2^{p-1}),$$

$$A_1(k2^q+j+2^{q-1}) = [A_2(k2^{q-1}+j) - A_2(k2^{q-1}+j+2^{q-1})]w(k2^{q-1}).$$

k,j 循环结束，做下一步；

步骤 7 若 $q=p$ 转步骤 8，否则 $q+1\to q$ 转步骤 4.

步骤 8 q 循环结束，若 $p=$偶数，将 $A_1(j)\to A_2(j)$，则 $c_j=A_2(j)(j=0,1,\cdots,N-1)$即为所求.

3.7 有理逼近

3.7.1 有理逼近与连分式

前面讨论了用多项式逼近函数 $f(x) \in C[a,b]$，多项式是一种计算简便的函数类，但当函数在某点附近无界时用多项式逼近效果很差，而用有理函数逼近则可得到较好的效果. 所谓有理函数逼近是指用形如

$$R_{nm}(x) = \frac{P_n(x)}{Q_m(x)} = \frac{\sum_{k=0}^{n} a_k x^k}{\sum_{k=0}^{m} b_k x^k} \tag{7.1}$$

的函数逼近 $f(x)$，与前面讨论一样，如果 $\| f(x) - R_{nm}(x) \|_\infty$ 最小就可得到**最佳有理一致逼近**，如果 $\| f(x) - R_{nm}(x) \|_2$ 最小则可得到**最佳有理平方逼近函数**. 这里不做具体介绍，可参看文献[5]. 本节主要讨论利用函数的泰勒展开获得有理逼近函数的方法. 先看例题，对函数 $\ln(1+x)$ 用泰勒展开得

$$\ln(1+x) = \sum_{k=1}^{\infty} (-1)^{k-1} \frac{x^k}{k} \qquad x \in [-1,1]. \tag{7.2}$$

取部分和

$$S_n(x) = \sum_{k=1}^{n} (-1)^{k-1} \frac{x^k}{k} \approx \ln(1+x).$$

另方面若对(7.2)式用辗转相除可得到 $\ln(1+x)$ 的一种连分式展开

$$\ln(1+x) = \cfrac{x}{1 + \cfrac{1 \cdot x}{2 + \cfrac{1 \cdot x}{3 + \cfrac{2^2 \cdot x}{4 + \cfrac{2^2 \cdot x}{5 + \cdots}}}}}$$

$$= \frac{x}{1} + \frac{1 \cdot x}{2} + \frac{1 \cdot x}{3} + \frac{2^2 \cdot x}{4} + \frac{2^2 \cdot x}{5} + \cdots. \tag{7.3}$$

(7.3)右端为 $\ln(1+x)$ 的无穷连分式的前 5 项，最后式子是它的紧凑形式，若取(7.3)的前 2,4,6,8 项，则可分别得到 $\ln(1+x)$ 的以下有理逼近

$$\begin{cases} R_{11}(x) = \dfrac{2x}{2+x}, \qquad R_{22}(x) = \dfrac{6x+3x^2}{6+6x+x^2}, \\ R_{33}(x) = \dfrac{60x+60x^2+11x^3}{60+90x+36x^2+3x^3}, \\ R_{44}(x) = \dfrac{420x+630x^2+260x^3+25x^4}{420+840x+540x^2+120x^3+6x^4}. \end{cases} \tag{7.4}$$

若用同样多项的泰勒展开部分和 $S_{2n}(x)$ 逼近 $\ln(1+x)$，并计算 $x=1$ 处的值 $S_{2n}(1)$ 及 $R_{nn}(1)$，计算结果见表 3-3.

表 3-3

n	$S_{2n}(1)$	$\varepsilon_S = \lvert \ln 2 - S_{2n}(1) \rvert$	$R_{nn}(1)$	$\varepsilon_R = \lvert \ln 2 - R_{nn}(1) \rvert$
1	0.50	0.19	0.667	0.026
2	0.58	0.11	0.69231	0.00084
3	0.617	0.076	0.693122	0.000025
4	0.634	0.058	0.69314642	0.00000076

ln2 的准确值为 0.69314718…，由此看出 $R_{44}(1)$ 的精度比 $S_8(1)$ 高出近 10 万倍，而它们的计算量是相当的，这说明用有理逼近比多项式逼近好得多. 在计算机上计算有理函数(7.1)的值通常可转化为连分式，这样可以节省乘除法的计算次数.

例 9 给出有理函数

$$R_{43}(x) = \frac{2x^4+45x^3+381x^2+1353x+1511}{x^3+21x^2+157x+409}$$

用辗转相除法将它化为连分式并写成紧凑形式.

解 用辗转相除可逐步得到

$$R_{43}(x)=2x+3+\frac{4x^2+64x+284}{x^3-21x^2+157x+409}$$

$$=2x+3+\cfrac{4}{x+5+\cfrac{6(x+9)}{x^2+16x+71}}$$

$$=2x+3+\cfrac{4}{x+5+\cfrac{6}{x+7+\cfrac{8}{x+9}}}$$

$$=2x+3+\frac{4}{x+5}+\frac{6}{x+7}+\frac{8}{x+9}.$$

本例中用连分式计算 $R_{43}(x)$ 的值只需 3 次除法，1 次乘法和 7 次加法. 若直接用多项式计算的秦九韶算法则需 6 次乘法和 1 次除法及 7 次加法，可见将 $R_{nm}(x)$ 化成连分式可节省计算乘除法次数，对一般的有理函数(7.1)可转化为一个连分式

$$R_{nm}(x)=P_1(x)+\frac{c_2}{x-d_1}+\cdots+\frac{c_l}{x+d_l}.$$

它的乘除法运算只需 $\max(m,n)$ 次，而直接用有理函数(7.1)计算乘除法次数为 $n+m$ 次.

3.7.2　帕德逼近

利用函数 $f(x)$ 的泰勒展开可以得到它的有理逼近. 设 $f(x)$ 在 $x=0$ 的泰勒展开为

$$f(x)=\sum_{k=0}^{N}\frac{1}{k!}f^{(k)}(0)x^k+\frac{f^{(N+1)}(\xi)}{(N+1)!}x^{N+1}. \tag{7.5}$$

它的部分和记作

$$P(x)=\sum_{k=0}^{N}\frac{1}{k!}f^{(k)}(0)x^k=\sum_{k=0}^{N}c_kx^k. \tag{7.6}$$

定义 11　设 $f\in C^{N+1}(-a,a)$，$N=n+m$，如果有理函数

$$R_{nm}(x)=\frac{a_0+a_1x+\cdots+a_nx^n}{1+b_1x+\cdots+b_mx^m}=\frac{P_n(x)}{Q_m(x)}, \tag{7.7}$$

其中 $P_n(x)$,$Q_m(x)$无公因式,且满足条件

$$R_{nm}^{(k)}(0) = f^{(k)}(0) \qquad (k = 0,1,\cdots,N), \tag{7.8}$$

则称 $R_{nm}(x)$为函数 $f(x)$在 $x=0$ 处的(n,m)阶**帕德(Padé)逼近**,记作$R(n,m)$,简称 $R(n,m)$的帕德逼近.

根据定义,若令

$$h(x) = P(x)Q_m(x) - P_n(x),$$

则满足条件(7.8)等价于

$$h^{(k)}(0) = 0, \qquad k = 0,1,\cdots,N.$$

即

$$h^{(k)}(0) = (P(x)Q_m(x) - P_n(x))^{(k)}\Big|_{x=0} = 0, \quad k = 0,1,\cdots,N.$$

由于 $P_n^{(k)}(0)=k!a_k$,应用莱布尼兹求导公式得

$$(P(x)Q_m(x) - P_n(x))^{(k)}\Big|_{x=0} = k!\sum_{j=0}^{k} c_j b_{k-j} - k!a_k = 0$$

$$k = 0,1,\cdots,N,$$

这里 $c_j=\frac{1}{j!}f^{(j)}(0)$是由(7.6)得到的,上式两端除 $k!$,并由 $b_0=1$,$b_j=0$(当 $j>m$ 时),可得

$$a_k = \sum_{j=0}^{k-1} c_j b_{k-j} + c_k \qquad (k = 0,1,\cdots,n) \tag{7.9}$$

及

$$-\sum_{j=0}^{k-1} c_j b_{k-j} = c_k \qquad (k = n+1,\cdots,n+m). \tag{7.10}$$

注意当 $j>m$ 时 $b_j=0$,故(7.10)可写成

$$\begin{cases} -c_{n-m+1}b_m - \cdots - c_{n-1}b_2 - c_n b_1 = c_{n+1}, \\ -c_{n-m+2}b_m - \cdots - c_n b_2 - c_{n+1}b_1 = c_{n+2}, \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ -c_n b_m - \cdots - c_{n+m-2}b_2 - c_{n+m-1}b_1 = c_{n+m}. \end{cases} \tag{7.11}$$

其中 $j<0$ 时 $c_j=0$,若记

$$
\boldsymbol{H} = \begin{pmatrix} -c_{n-m+1} & \cdots & -c_{n-1} & -c_n \\ -c_{n-m+2} & \cdots & -c_n & -c_{n+1} \\ \vdots & & \vdots & \vdots \\ -c_n & \cdots & -c_{n+m-2} & -c_{n+m-1} \end{pmatrix}, \qquad (7.12)
$$

$$
\bar{\boldsymbol{b}} = (b_m, b_{m-1}, \cdots, b_1)^{\mathrm{T}}, \quad \bar{\boldsymbol{c}} = (c_{n+1}, c_{n+2}, \cdots, c_{n+m})^{\mathrm{T}}.
$$

则方程组(7.11)的矩阵形式为

$$
\boldsymbol{H}\bar{\boldsymbol{b}} = \bar{\boldsymbol{c}}.
$$

综上所述得下面的定理.

定理 10　设 $f(x) \in C^{N+1}(-a, a)$，$N = n + m$，则形如(7.7)的有理函数 $R_{nm}(x)$ 是 $f(x)$ 的 (n, m) 阶帕德逼近的充分必要条件是多项式 $P_n(x)$ 及 $Q_m(x)$ 的系数 $a_0, a_1, \cdots, a_n$ 及 $b_1, \cdots, b_m$ 满足方程组(7.9)及(7.11).

根据定理 10 求 $f(x)$ 的帕德逼近时，首先要由(7.11)解出 $Q_m(x)$ 的系数 $b_1, \cdots, b_m$，再由(7.9)直接算出 $P_n(x)$ 的系数 $a_0, a_1, \cdots, a_n$. $f(x)$ 的各阶帕德逼近可列成一张表，称为帕德表(见表 3-4).

表 3-4　帕德表

$n \backslash m$	0	1	2	3	4	…
0	(0,0)	(0,1)	(0,2)	(0,3)	(0,4)	…
1	(1,0)	(1,1)	(1,2)	(1,3)	(1,4)	…
2	(2,0)	(2,1)	(2,2)	(2,3)	(2,4)	…
3	(3,0)	(3,1)	(3,2)	(3,3)	(3,4)	…
4	(4,0)	(4,1)	(4,2)	(4,3)	(4,4)	…
⋮	⋮	⋮	⋮	⋮	⋮	

例 10　求 $f(x) = \ln(1 + x)$ 的帕德逼近 $R(2,2)$ 及 $R(3,3)$.

解 由 $\ln(1+x)$的泰勒展开

$$\ln(1+x)=x-\frac{1}{2}x^2+\frac{1}{3}x^3-\frac{1}{4}x^4+\cdots$$

得 $c_0=0,c_1=1,c_2=-\frac{1}{2},c_3=\frac{1}{3},c_4=-\frac{1}{4},\cdots$. 当 $n=m=2$ 时，由(7.11)得

$$\begin{cases}-b_2+\frac{1}{2}b_1=\frac{1}{3},\\ \frac{1}{2}b_2-\frac{1}{3}b_1=-\frac{1}{4}.\end{cases}$$

求得 $b_1=1,b_2=\frac{1}{6}$，再由(7.9)得

$$a_0=0,\ a_1=1,\ a_2=\frac{1}{2},$$

于是得

$$R_{22}(x)=\frac{x+\frac{1}{2}x^2}{1+x+\frac{1}{6}x^2}=\frac{6x+3x^2}{6+6x+x^2}.$$

当 $n=m=3$ 时，由(7.11)得

$$\begin{cases}-b_3+\frac{1}{2}b_2-\frac{1}{3}b_1=-\frac{1}{4},\\ \frac{1}{2}b_3-\frac{1}{3}b_2+\frac{1}{4}b_1=\frac{1}{5},\\ -\frac{1}{3}b_3+\frac{1}{4}b_2-\frac{1}{5}b_1=-\frac{1}{6}.\end{cases}$$

解得 $b_1=\frac{3}{2},\ b_2=\frac{3}{5},\ b_3=\frac{1}{20}.$

代入(7.9)得 $a_0=0,\ a_1=1,\ a_2=1,\ a_3=\frac{11}{60}.$

于是得

$$R_{33}(x) = \frac{x + x^2 + \frac{11}{60}x^3}{1 + \frac{3}{2}x + \frac{3}{5}x^2 + \frac{1}{20}x^3} = \frac{60x + 60x^2 + 11x^3}{60 + 90x + 36x^2 + 3x^3}.$$

可以看到这里得到的 $R_{22}(x)$ 及 $R_{33}(x)$ 与 $\ln(1+x)$ 的前面连分式展开得到的有理逼近(7.4)结果一样.

为了求帕德逼近 $R_{nm}(x)$ 的误差估计，由(7.9)及(7.11)求得的 $P_n(x)$，$Q_m(x)$ 系数 $a_0, a_1, \cdots, a_n$ 及 $b_1, \cdots, b_m$，直接代入则得

$$f(x)Q_m(x) - P_n(x) = x^{n+m+1}\Big(\sum_{l=0}^{\infty}\sum_{k=0}^{m} b_k c_{n+m+1+l-k}\Big)x^l,$$

将 $Q_m(x)$ 除上式两端，即得

$$f(x) - R_{nm}(x) = \frac{x^{n+m+1}\sum_{l=0}^{\infty} r_l x^l}{Q_m(x)}, \tag{7.13}$$

其中 $r_l = \sum_{k=0}^{m} b_k c_{n+m+l+1-k}$.

当 $|x|<1$ 时可得误差近似表达式

$$f(x) - R_{nm}(x) \approx r_0 x^{n+m+1}, \quad r_0 = \sum_{k=0}^{m} b_k c_{n+m+1-k}.$$

评 注

函数逼近是数学中的经典课题，它与数学中其他分支有着密切的联系，也是计算数学的基础，本章仅讨论最佳一致逼近和最佳平方逼近的基本概念及正交多项式，更详细内容可参看[5][6]等专门著作.

正交多项式在函数逼近中有重要作用，它在高斯(Gauss)求积中也有重要应用. 特别地，对勒让德多项式及切比雪夫多项式本章做了较详细的讨论，因为这是两个十分重要又经常使用的正

交多项式，应引起读者关注.

曲线拟合的最小二乘法在应用科学中具有重要作用，它是离散点的最佳平方逼近. 本章引入哈尔条件，由此可证明解的存在唯一性，而采用离散点正交多项式可避免解法方程时出现的病态问题，为用多项式作最小二乘模型提供了可行的算法.

傅里叶变换也是最佳平方逼近得到的，快速傅里叶变换(FFT)是节省计算次数的一个范例，本章介绍的 FFT 算法是 Cooley-Tukey 算法的改进[12]，其基本思想与经典的 FFT 算法是一致的，更详细讨论可参见文献[11].

有理逼近是函数逼近的重要组成部分. 本章只简单介绍有广泛应用的帕德逼近，其他内容可参见文献[5].

习　题

1. $f(x)=\sin\frac{\pi}{2}x$，给出[0,1]上的伯恩斯坦多项式 $B_1(f,x)$ 及 $B_3(f,x)$.

2. 当 $f(x)=x$ 时，求证 $B_n(f,x)=x$.

3. 证明函数 $1,x,\cdots,x^n$ 线性无关.

4. 计算下列函数 $f(x)$ 关于 $C[0,1]$ 的 $\|f\|_\infty$，$\|f\|_1$ 与 $\|f\|_2$：

(1) $f(x)=(x-1)^3$，

(2) $f(x)=\left|x-\frac{1}{2}\right|$，

(3) $f(x)=x^m(1-x)^n$，m 与 n 为正整数，

(4) $f(x)=(x+1)^{10}e^{-x}$.

5. 证明 $\|f-g\|\geqslant\|f\|-\|g\|$.

6. 对 $f(x),g(x)\in C^1[a,b]$，定义

(1) $(f,g)=\int_a^b f'(x)g'(x)dx$，

(2) $(f,g)=\int_a^b f'(x)g'(x)dx+f(a)g(a)$.

问它们是否构成内积.

7. 令 $T_n^*(x)=T_n(2x-1), x\in[0,1]$,试证$\{T_n^*(x)\}$是在$[0,1]$上带权 $\rho(x)=\dfrac{1}{\sqrt{x-x^2}}$的正交多项式,并求 $T_0^*(x), T_1^*(x), T_2^*(x), T_3^*(x)$.

8. 对权函数 $\rho(x)=1+x^2$,区间$[-1,1]$,试求首项系数为 1 的正交多项式 $\varphi_n(x), n=0,1,2,3$.

9. 试证明由(2.14)给出的第二类切比雪夫多项式族$\{u_n(x)\}$是$[-1,1]$上带权 $\rho(x)=\sqrt{1-x^2}$的正交多项式.

10. 证明切比雪夫多项式 $T_n(x)$满足微分方程

$$(1-x^2)T_n''(x)-xT_n'(x)+n^2T_n(x)=0.$$

11. 假设 $f(x)$在$[a,b]$上连续,求 $f(x)$的零次最佳一致逼近多项式.

12. 选取常数 a,使 $\max\limits_{0\leqslant x\leqslant 1}|x^3-ax|$达到极小,又问这个解是否唯一?

13. 求 $f(x)=\sin x$ 在$[0,\pi/2]$上的最佳一次逼近多项式,并估计误差.

14. 求 $f(x)=e^x$ 在$[0,1]$上的最佳一次逼近多项式.

15. 求 $f(x)=x^4+3x^3-1$ 在区间$[0,1]$上的三次最佳一致逼近多项式.

16. $f(x)=|x|$,在$[-1,1]$上求关于 $\Phi=\text{span}\{1,x^2,x^4\}$的最佳平方逼近多项式.

17. 求函数 $f(x)$在指定区间上对于 $\Phi=\text{span}\{1,x\}$的最佳平方逼近多项式:

(1) $f(x)=\dfrac{1}{x},[1,3]$;　　　　(2) $f(x)=e^x,[0,1]$;

(3) $f(x)=\cos\pi x,[0,1]$;　　　　(4) $f(x)=\ln x,[1,2]$.

18. $f(x)=\sin\dfrac{\pi}{2}x$,在$[-1,1]$上按勒让德多项式展开求三次最佳平方逼近多项式.

19. 观测物体的直线运动,得出以下数据:

时间 t(s)	0	0.9	1.9	3.0	3.9	5.0
距离 s(m)	0	10	30	50	80	110

求运动方程.

20. 已知实验数据如下:

x_i	19	25	31	38	44
y_i	19.0	32.3	49.0	73.3	97.8

用最小二乘法求形如 $y=a+bx^2$ 的经验公式,并计算均方误差.

21. 在某化学反应中,由实验得分解物浓度与时间关系如下:

时间 t	0	5	10	15	20	25	30	35	40	45	50	55
浓度 $y(\times 10^{-4})$	0	1.27	2.16	2.86	3.44	3.87	4.15	4.37	4.51	4.58	4.62	4.64

用最小二乘法求 $y=f(t)$.

22. 给出一张记录$\{f_k\}=(4,3,2,1,0,1,2,3)$,用 FFT 算法求$\{f_k\}$的离散谱$\{c_k\}$.

23. 用辗转相除法将 $R_{22}(x)=\dfrac{3x^2+6x}{x^2+6x+6}$化为连分式.

24. 求 $f(x)=\sin x$ 在 $x=0$ 处的(3,3)阶帕德逼近 $R_{33}(x)$.

25. 求 $f(x)=e^x$ 在 $x=0$ 处的(2,1)阶帕德逼近 $R_{21}(x)$.

第 4 章 数值积分与数值微分

4.1 引 言

4.1.1 数值求积的基本思想

实际问题当中常常需要计算积分. 有些数值方法,如微分方程和积分方程的求解,也都和积分计算相联系.

依据人们所熟知的微积分基本定理,对于积分

$$I=\int_a^b f(x)\mathrm{d}x,$$

只要找到被积函数 $f(x)$的原函数 $F(x)$,便有下列牛顿-莱布尼兹(Newton-Leibniz)公式:

$$\int_a^b f(x)\mathrm{d}x=F(b)-F(a).$$

但实际使用这种求积方法往往有困难,因为大量的被积函数,诸如 $\frac{\sin x}{x}$, $\sin x^2$ 等等,找不到用初等函数表示的原函数;另外,当 $f(x)$是由测量或数值计算给出的一张数据表时,牛顿-莱布尼兹公式也不能直接运用. 因此有必要研究积分的数值计算问题.

积分中值定理告诉我们,在积分区间$[a,b]$内存在一点 ξ,成立

$$\int_a^b f(x)\mathrm{d}x=(b-a)f(\zeta),$$

就是说,底为 $b-a$ 而高为 $f(\zeta)$的矩形面积恰等于所求曲边梯形的面积 I(图 4-1). 问题在于点 ξ 的具体位置一般是不知道的,因而难以准确算出 $f(\zeta)$的值. 我们将 $f(\zeta)$称为区间$[a,b]$上的平均

高度. 这样,只要对平均高度 $f(\zeta)$ 提供一种算法,相应地便获得一种数值求积方法.

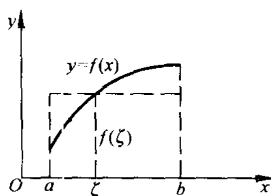


图 4-1

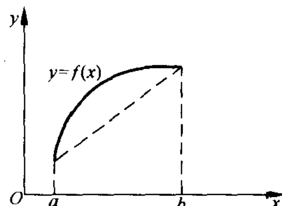


图 4-2

如果我们用两端点"高度"$f(a)$与 $f(b)$的算术平均作为平均高度 $f(\xi)$的近似值,这样导出的求积公式

$$T = \frac{b-a}{2}[f(a)+f(b)] \tag{1.1}$$

便是我们所熟悉的**梯形公式**(几何意义参看图 4-2). 而如果改用区间中点 $c=\dfrac{a+b}{2}$的"高度"$f(c)$近似地取代平均高度 $f(\zeta)$,则又可导出所谓**中矩形公式**(今后简称**矩形公式**)

$$R = (b-a)f\left(\frac{a+b}{2}\right). \tag{1.2}$$

更一般地,我们可以在区间$[a,b]$上适当选取某些节点 x_k,然后用 $f(x_k)$加权平均得到平均高度 $f(\zeta)$的近似值,这样构造出的求积公式具有下列形式:

$$\int_a^b f(x)\mathrm{d}x \approx \sum_{k=0}^{n} A_k f(x_k), \tag{1.3}$$

式中 x_k 称为**求积节点**;A_k 称为**求积系数**,亦称伴随节点 x_k 的**权**. 权 A_k 仅仅与节点 x_k 的选取有关,而不依赖于被积函数 $f(x)$的具体形式.

这类数值积分方法通常称为**机械求积**,其特点是将积分求值

问题归结为函数值的计算，这就避开了牛顿-莱布尼兹公式需要寻求原函数的困难.

4.1.2 代数精度的概念

数值求积方法是近似方法，为要保证精度，我们自然希望求积公式能对"尽可能多"的函数准确地成立，这就提出了所谓代数精度的概念.

定义1 如果某个求积公式对于次数不超过 m 的多项式均能准确地成立，但对于 $m+1$ 次多项式就不准确成立，则称该求积公式具有 **m 次代数精度.**

不难验证，梯形公式(1.1)和矩形公式(1.2)均具有一次代数精度.

一般地，欲使求积公式(1.3)具有 m 次代数精度，只要令它对于 $f(x)=1,x,\cdots,x^m$ 都能准确成立，这就要求

$$\begin{cases}\sum A_k = b-a,\\ \sum A_k x_k = \dfrac{1}{2}(b^2-a^2),\\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\ \sum A_k x_k^m = \dfrac{1}{m+1}(b^{m+1}-a^{m+1}).\end{cases}\tag{1.4}$$

为简洁起见，这里省略了符号 $\sum\limits_{k=0}^{n}$ 中的上下标.

如果我们事先选定求积节点 x_k，譬如，以区间$[a,b]$的等距分点作为节点，这时取 $m=n$ 求解方程组(1.4)即可确定求积系数 A_k，而使求积公式(1.3)至少具有 n 次代数精度. 本章第2节介绍这样一类求积公式，梯形公式是其中的一个特例.

为了构造出形如(1.3)的求积公式，原则上是一个确定参数 x_k 和 A_k 的代数问题.

4.1.3 插值型的求积公式

设给定一组节点

$$a \leqslant x_0 < x_1 < x_2 < \cdots < x_n \leqslant b,$$

且已知函数 $f(x)$在这些节点上的值,作插值函数 $L_n(x)$(参看第 2 章(2.9)式). 由于代数多项式 $L_n(x)$的原函数是容易求出的,我们取

$$I_n = \int_a^b L_n(x)\mathrm{d}x$$

作为积分 $I = \int_a^b f(x)\mathrm{d}x$ 的近似值,这样构造出的求积公式

$$I_n = \sum_{k=0}^{n} A_k f(x_k) \tag{1.5}$$

称为是**插值型**的,式中求积系数 A_k 通过插值基函数 $l_k(x)$积分得出

$$A_k = \int_a^b l_k(x)\mathrm{d}x. \tag{1.6}$$

由插值余项定理(第 2 章的定理 2)即知,对于插值型的求积公式(1.5),其余项

$$R[f] = I - I_n = \int_a^b \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega(x)\mathrm{d}x, \tag{1.7}$$

式中 ξ 与变量 x 有关,$\omega(x)=(x-x_0)(x-x_1)\cdots(x-x_n)$.

如果求积公式(1.5)是插值型的,按式(1.7),对于次数不超过 n 的多项式 $f(x)$,其余项 $R[f]$等于零,因而这时求积公式至少具有 n 次代数精度.

反之,如果求积公式(1.5)至少具有 n 次代数精度,则它必定是插值型的. 事实上,这时公式(1.5)对于插值基函数 $l_k(x)$应准确成立,即有

$$\int_a^b l_k(x)\mathrm{d}x = \sum_{j=0}^{n} A_j l_k(x_j).$$

注意到 $l_k(x_j)=\delta_{kj}$,上式右端实际上即等于 A_k,因而式(1.6)

成立.

综上所述,我们的结论是

定理 1　形如(1.5)的求积公式至少有 n 次代数精度的充分必要条件是,它是插值型的.

4.1.4　求积公式的收敛性与稳定性

定义 2　在求积公式(1.3)中,若

$$\lim_{\substack{n\to\infty \\ h\to 0}} \sum_{k=0}^{n} A_k f(x_k) = \int_a^b f(x)\mathrm{d}x.$$

其中 $h=\max\limits_{1\leqslant i\leqslant n}(x_i-x_{i-1})$,则称求积公式(1.3)是收敛的.

在求积公式(1.3)中,由于计算 $f(x_k)$ 可能产生误差 δ_k,实际得到 $\tilde{f}_k$,即 $f(x_k)=\tilde{f}_k+\delta_k$. 记

$$I_n(f) = \sum_{k=0}^{n} A_k f(x_k), \quad I_n(\tilde{f}) = \sum_{k=0}^{n} A_k \tilde{f}_k.$$

如果对任给小正数 $\varepsilon>0$,只要误差 $|\delta_k|$ 充分小就有

$$|I_n(f)-I_n(\tilde{f})| = \left|\sum_{k=0}^{n} A_k[f(x_k)-\tilde{f}_k]\right| \leqslant \varepsilon, \tag{1.8}$$

它表明求积公式(1.3)计算是稳定的,由此给出:

定义 3　对任给 $\varepsilon>0$,若 $\exists\,\delta>0$,只要 $|f(x_k)-\tilde{f}_k|\leqslant\delta(k=0,1,\cdots,n)$ 就有(1.8)成立,则称求积公式(1.3)是稳定的.

定理 2　若求积公式(1.3)中系数 $A_k>0(k=0,1,\cdots,n)$,则此求积公式是稳定的.

证明　对任给 $\varepsilon>0$,若取 $\delta=\dfrac{\varepsilon}{b-a}$,对 $k=0,1,\cdots,n$ 都有 $|f(x_k)-\tilde{f}_k|\leqslant\delta$,则有

$$\begin{aligned}|I_n(f)-I_n(\tilde{f})| &= \left|\sum_{k=0}^{n} A_k(f(x_k)-\tilde{f}_k)\right| \\ &\leqslant \sum_{k=0}^{n} |A_k|\,|f(x_k)-\tilde{f}_k|\end{aligned}$$

$$\leqslant \delta \sum_{k=0}^{n} A_k = \delta(b-a) = \varepsilon.$$

由定义 3 可知求积公式(1.3)是稳定的. 证毕.

定理 2 表明只要求积系数 $A_k>0$,就能保证计算的稳定性.

4.2 牛顿-柯特斯公式

4.2.1 柯特斯系数

设将积分区间$[a,b]$划分为 n 等分,步长 $h=\dfrac{b-a}{n}$,选取等距节点 $x_k=a+kh$ 构造出的插值型求积公式

$$I_n = (b-a)\sum_{k=0}^{n} C_k^{(n)} f(x_k) \tag{2.1}$$

称为**牛顿-柯特斯**(Newton-Cotes)**公式**,式中 $C_k^{(n)}$ 称为**柯特斯系数**. 按(1.6)式,引进变换 $x=a+th$,则有

$$\begin{aligned} C_k^{(n)} &= \frac{h}{b-a}\int_0^n \prod_{\substack{j=0\\ j\neq k}}^{n} \frac{t-j}{k-j}\mathrm{d}t \\ &= \frac{(-1)^{n-k}}{nk!(n-k)!}\int_0^n \prod_{\substack{j=0\\ j\neq k}}^{n} (t-j)\mathrm{d}t. \end{aligned} \tag{2.2}$$

由于是多项式的积分,柯特斯系数的计算不会遇到实质性的困难. 当 $n=1$ 时,

$$C_0^{(1)} = C_1^{(1)} = \frac{1}{2},$$

这时的求积公式就是我们所熟悉的梯形公式(1.1).

当 $n=2$ 时,按(2.2)式,这时柯特斯系数为

$$C_0^{(2)} = \frac{1}{4}\int_0^2 (t-1)(t-2)\mathrm{d}t = \frac{1}{6},$$

$$C_1^{(2)} = -\frac{1}{2}\int_0^2 t(t-2)\mathrm{d}t = \frac{4}{6},$$

$$C_2^{(2)} = \frac{1}{4}\int_0^2 t(t-1)\mathrm{d}t = \frac{1}{6}.$$

相应的求积公式是下列**辛普森**(Simpson)**公式**

$$S = \frac{b-a}{6}\left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right], \tag{2.3}$$

而 $n=4$ 的牛顿-柯特斯公式则特别称为**柯特斯公式**,其形式是

$$C = \frac{b-a}{90}[7f(x_0) + 32f(x_1) + 12f(x_2) + 32f(x_3) + 7f(x_4)]. \tag{2.4}$$

这里 $x_k = a + kh, h = \frac{b-a}{4}$.

下表列出柯特斯系数表开头的一部分.

表　4-1

n	$C_k^{(n)}$								
1	$\frac{1}{2}$	$\frac{1}{2}$							
2	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$						
3	$\frac{1}{8}$	$\frac{3}{8}$	$\frac{3}{8}$	$\frac{1}{8}$					
4	$\frac{7}{90}$	$\frac{16}{45}$	$\frac{2}{15}$	$\frac{16}{45}$	$\frac{7}{90}$				
5	$\frac{19}{288}$	$\frac{25}{96}$	$\frac{25}{144}$	$\frac{25}{144}$	$\frac{25}{96}$	$\frac{19}{288}$			
6	$\frac{41}{840}$	$\frac{9}{35}$	$\frac{9}{280}$	$\frac{34}{105}$	$\frac{9}{280}$	$\frac{9}{35}$	$\frac{41}{840}$		
7	$\frac{751}{17280}$	$\frac{3577}{17280}$	$\frac{1323}{17280}$	$\frac{2989}{17280}$	$\frac{2989}{17280}$	$\frac{1323}{17280}$	$\frac{3577}{17280}$	$\frac{751}{17280}$	
8	$\frac{989}{28350}$	$\frac{5888}{28350}$	$\frac{-928}{28350}$	$\frac{10496}{28350}$	$\frac{-4540}{28350}$	$\frac{10496}{28350}$	$\frac{-928}{28350}$	$\frac{5888}{28350}$	$\frac{989}{28350}$

从表中看到 $n \geqslant 8$ 时,柯特斯系数 $C_k^{(n)}$ 出现负值,于是有

$$\sum_{k=0}^{n} | C_k^{(n)} | > \sum_{k=0}^{n} C_k^{(n)} = 1,$$

特别地,假定 $C_k^{(n)}(f(x_k)-\tilde{f}_k)>0$,且$|f(x_k)-\tilde{f}_k|=\delta$,则有

$$\begin{aligned}
| I_n(f) - I_n(\tilde{f}) | &= \left| \sum_{k=0}^{n} C_k^{(n)} [f(x_k) - \tilde{f}_k] \right| \\
&= \sum_{k=0}^{n} C_k^{(n)} [f(x_k) - \tilde{f}_k] \\
&= \sum_{k=0}^{n} | C_k^{(n)} | | f(x_k) - \tilde{f}_k | = \delta \sum_{k=0}^{n} | C_k^{(n)} | > \delta.
\end{aligned}$$

它表明初始数据误差将会引起计算结果误差增大,即计算不稳定,故 $n \geqslant 8$ 的牛顿-柯特斯公式是不用的.

4.2.2 偶阶求积公式的代数精度

作为插值型的求积公式,n 阶的牛顿-柯特斯公式至少具有 n 次的代数精度(定理 1). 实际的代数精度能否进一步提高呢?

先看辛普森公式(2.3),它是二阶牛顿-柯特斯公式,因此至少具有二次代数精度. 进一步用 $f(x)=x^3$ 进行检验,按辛普森公式计算得

$$S = \frac{b-a}{6}\left[a^3 - 4\left(\frac{a+b}{2}\right)^3 + b^3\right].$$

另一方面,直接求积得

$$I = \int_a^b x^3 \mathrm{d}x = \frac{b^4 - a^4}{4}.$$

这时有 $S=I$,即辛普森公式对次数不超过三次的多项式均能准确成立,又容易验证它对 $f(x)=x^4$ 通常是不准确的,因此,辛普森公式实际上具有三次代数精度.

一般地,我们可以证明下述论断:

定理 3 当阶 n 为偶数时,牛顿-柯特斯公式(2.1)至少有 $n+1$ 次代数精度.

证明 我们只要验证，当 n 为偶数时，牛顿-柯特斯公式对 $f(x)=x^{n+1}$ 的余项为零.

按余项公式(1.7)，由于这里 $f^{(n+1)}(x)=(n+1)!$，从而有

$$R[f]=\int_a^b \prod_{j=0}^{n}(x-x_j)\mathrm{d}x.$$

引进变换 $x=a+th$，并注意到 $x_j=a+jh$，有

$$R[f]=h^{n+2}\int_0^n \prod_{j=0}^{n}(t-j)\mathrm{d}t,$$

若 n 为偶数，则 $\frac{n}{2}$ 为整数，再令 $t=u+\frac{n}{2}$，进一步有

$$R[f]=h^{n+2}\int_{-\frac{n}{2}}^{\frac{n}{2}} \prod_{j=0}^{n}\left(u+\frac{n}{2}-j\right)\mathrm{d}u,$$

据此可以断定 $R[f]=0$，因为被积函数

$$H(u)=\prod_{j=0}^{n}\left(u+\frac{n}{2}-j\right)=\prod_{j=-n/2}^{n/2}(u-j)$$

是个奇函数. 证毕.

4.2.3 几种低阶求积公式的余项

首先考察梯形公式，按余项公式(1.7)，梯形公式(1.1)的余项

$$R_T=I-T=\int_a^b \frac{f''(\xi)}{2}(x-a)(x-b)\mathrm{d}x,$$

这里积分的核函数 $(x-a)(x-b)$ 在区间 $[a,b]$ 上保号(非正)，应用积分中值定理，在 $[a,b]$ 内存在一点 η，使

$$\begin{aligned}R_T&=\frac{f''(\eta)}{2}\int_a^b (x-a)(x-b)\mathrm{d}x\\&=-\frac{f''(\eta)}{12}(b-a)^3,\quad \eta\in[a,b].\end{aligned}\tag{2.5}$$

再研究辛普森公式(2.3)的余项 $R_S=I-S$. 为此构造次数不超过 3 的多项式 $H(x)$，使满足

$$H(a)=f(a),\quad H(b)=f(b);$$

$$H(c) = f(c), \quad H'(c) = f'(c). \tag{2.6}$$

这里 $c=\frac{a+b}{2}$. 由于辛普森公式具有三次代数精度,它对于这样构造出的三次式 $H(x)$是准确的,即

$$\int_a^b H(x)\mathrm{d}x = \frac{b-a}{6}[H(a) + 4H(c) + H(b)],$$

而利用插值条件(2.6)知,上式右端实际上等于按辛普森公式(2.3)求得的积分值 S,因此积分余项

$$R_S = I - S = \int_a^b [f(x) - H(x)]\mathrm{d}x.$$

对于满足条件(2.6)的多项式 $H(x)$,其插值余项由第 2 章(5.11)得

$$f(x) - H(x) = \frac{f^{(4)}(\xi)}{4!}(x-a)(x-c)^2(x-b),$$

故有

$$R_S = \int_a^b \frac{f^{(4)}(\xi)}{4!}(x-a)(x-c)^2(x-b)\mathrm{d}x.$$

这时积分的核函数$(x-a)(x-c)^2(x-b)$在$[a,b]$上保号(非正),再用积分中值定理有

$$\begin{aligned} R_S &= \frac{f^{(4)}(\eta)}{4!}\int_a^b (x-a)(x-c)^2(x-b)\mathrm{d}x \\ &= -\frac{b-a}{180}\left(\frac{b-a}{2}\right)^4 f^{(4)}(\eta). \end{aligned} \tag{2.7}$$

关于柯特斯公式(2.4)的积分余项,这里不再具体推导,仅列出结果如下:

$$R_C = I - C = -\frac{2(b-a)}{945}\left(\frac{b-a}{4}\right)^6 f^{(6)}(\eta). \tag{2.8}$$

4.3　复化求积公式

前面已经指出高阶牛顿-柯特斯公式是不稳定的,因此,不可

能通过提高阶的方法来提高求积精度. 为了提高精度通常可把积分区间分成若干子区间(通常是等分),再在每个子区间上用低阶求积公式. 这种方法称为复化求积法. 本节只讨论复化梯形公式与复化辛普森公式.

4.3.1　复化梯形公式

将区间$[a,b]$划分为 n 等分,分点 $x_k=kh$, $h=\frac{b-a}{n}$, $k=0,1,\cdots,n$,在每个子区间$[x_k,x_{k+1}]$($k=0,1,\cdots,n-1$)上采用梯形公式(1.1),则得

$$I=\int_a^b f(x)\mathrm{d}x=\sum_{k=0}^{n-1}\int_{x_k}^{x_{k+1}}f(x)\mathrm{d}x$$
$$=\frac{h}{2}\sum_{k=0}^{n-1}[f(x_k)+f(x_{k+1})]+R_n(f). \quad (3.1)$$

记

$$T_n=\frac{h}{2}\sum_{k=0}^{n-1}[f(x_k)+f(x_{k+1})]$$
$$=\frac{h}{2}\left[f(a)+2\sum_{k=1}^{n-1}f(x_k)+f(b)\right], \quad (3.2)$$

称为**复化梯形公式**,其余项可由(2.5)得

$$R_n(f)=I-T_n=\sum_{k=0}^{n-1}\left[-\frac{h^3}{12}f''(\eta_k)\right],\ \eta_k\in(x_k,x_{k+1}).$$

由于 $f(x)\in C^2[a,b]$,且

$$\min_{0\leqslant k\leqslant n-1}f''(\eta_k)\leqslant\frac{1}{n}\sum_{k=0}^{n-1}f''(\eta_k)\leqslant\max_{0\leqslant k\leqslant n-1}f''(\eta_k).$$

所以 $\exists\ \eta\in(a,b)$使

$$f''(\eta)=\frac{1}{n}\sum_{k=0}^{n-1}f''(\eta_k).$$

于是复化梯形公式余项为

$$R_n(f)=-\frac{b-a}{12}h^2 f''(\eta). \tag{3.3}$$

可以看出误差是 h^2 阶，且由(3.3)立即得到，当 $f(x)\in C^2[a,b]$，则

$$\lim_{n\to\infty}T_n=\int_a^b f(x)\mathrm{d}x,$$

即复化梯形公式是收敛的. 事实上只要设 $f(x)\in C[a,b]$，则可得到收敛性，因为只要把 T_n 改写为

$$T_n=\frac{1}{2}\left[\frac{b-a}{n}\sum_{k=0}^{n-1}f(x_k)+\frac{b-a}{n}\sum_{k=1}^{n}f(x_k)\right].$$

当 $n\to\infty$ 时，上式右端括号内的两个和式均收敛到积分 $\int_a^b f(x)\mathrm{d}x$，所以复化梯形公式(3.2)收敛. 此外，T_n 的求积系数为正，由定理 2 知复化梯形公式是稳定的.

4.3.2 复化辛普森求积公式

将区间$[a,b]$分为 n 等分，在每个子区间$[x_k,x_{k+1}]$上采用辛普森公式(2.3)，若记 $x_{k+1/2}=x_k+\frac{1}{2}h$，则得

$$\begin{aligned}I=\int_a^b f(x)\mathrm{d}x&=\sum_{k=0}^{n-1}\int_{x_k}^{x_{k+1}}f(x)\mathrm{d}x\\&=\frac{h}{6}\sum_{k=0}^{n-1}[f(x_k)+4f(x_{k+1/2})+f(x_{k+1})]+R_n(f).\end{aligned} \tag{3.4}$$

记

$$\begin{aligned}S_n&=\frac{h}{6}\sum_{k=0}^{n-1}[f(x_k)+4f(x_{k+1/2})+f(x_{k+1})]\\&=\frac{h}{6}\left[f(a)+4\sum_{k=0}^{n-1}f(x_{k+1/2})+2\sum_{k=1}^{n-1}f(x_k)+f(b)\right],\end{aligned} \tag{3.5}$$

称为**复化辛普森求积公式**. 其余项由(2.7)得

$$R_n(f) = I - S_n = -\frac{h}{180}\left(\frac{h}{2}\right)^4 \sum_{k=0}^{n-1} f^{(4)}(\eta_k), \quad \eta_k \in (x_k, x_{k+1}),$$

于是当 $f(x) \in C^4[a,b]$时，与复化梯形公式相似有

$$R_n(f) = I - S_n = -\frac{b-a}{180}\left(\frac{h}{2}\right)^4 f^{(4)}(\eta), \qquad \eta \in (a,b). \tag{3.6}$$

由(3.6)看出，误差阶为 h^4，收敛性是显然的，实际上，只要 $f(x) \in C[a,b]$则可得到收敛性，即

$$\lim_{n\to\infty} S_n = \int_a^b f(x)\mathrm{d}x.$$

此外，由于 S_n 中求积系数均为正数，故知复化辛普森公式计算稳定.

例 1 对于函数 $f(x) = \frac{\sin x}{x}$，给出 $n = 8$ 的函数表(见表 4-2)，试用复化梯形公式(3.2)及复化辛普森公式(3.5)计算积分

$$I = \int_0^1 \frac{\sin x}{x}\mathrm{d}x,$$

并估计误差.

解 将积分区间[0,1]划分为 8 等分，应用复化梯形法求得

$$T_8 = 0.9456909;$$

而如果将[0,1]分为 4 等分，应用复化辛普森法有

$$S_4 = 0.9460832.$$

比较上面两个结果 T_8 与 S_4，它们都需要提供 9 个点上的函数值，计算量基本相同，然而精度却差别很大，同积分的准确值 $I = 0.9460831$ 比较，复化梯形法的结

表 4-2

x	$f(x)$
0	1
1/8	0.9973978
1/4	0.9896158
3/8	0.9767267
1/2	0.9588510
5/8	0.9361556
3/4	0.9088516
7/8	0.8771925
1	0.8414709

果 $T_8 = 0.9456909$ 只有两位有效数字，而复化辛普森法的结果 $S_4 = 0.9460832$ 却有六位有效数字.

为了利用余项公式估计误差，要求 $f(x)=\dfrac{\sin x}{x}$ 的高阶导数，由于

$$f(x)=\frac{\sin x}{x}=\int_0^1 \cos(xt)\mathrm{d}t,$$

所以有

$$f^{(k)}(x)=\int_0^1 \frac{\mathrm{d}^k}{\mathrm{d}x^k}(\cos xt)\mathrm{d}t=\int_0^1 t^k \cos\left(xt+\frac{k\pi}{2}\right)\mathrm{d}t,$$

于是

$$\max_{0\leqslant x\leqslant 1}|f^{(k)}(x)|\leqslant\int_0^1\left|\cos\left(xt+\frac{k\pi}{2}\right)\right|t^k\mathrm{d}t\leqslant\int_0^1 t^k\mathrm{d}t=\frac{1}{k+1}.$$

由(3.3)得复化梯形公式误差

$$|R_8(f)|=|I-T_8|\leqslant\frac{h^2}{12}\max_{0\leqslant x\leqslant 1}|f''(x)|$$

$$\leqslant\frac{1}{12}\left(\frac{1}{8}\right)^2\frac{1}{3}=0.000434.$$

对复化辛普森公式误差，由(3.6)得

$$|R_4(f)|=|I-S_4|\leqslant\frac{1}{2880}\left(\frac{1}{4}\right)^4\frac{1}{5}=0.271\times 10^{-6}.$$

4.4 龙贝格求积公式

4.4.1 梯形法的递推化

上节介绍的复化求积方法可提高求积精度，实际计算时若精度不够可将步长逐次分半. 设将区间$[a,b]$分为 n 等分，共有 $n+1$ 个分点，如果将求积区间再二分一次，则分点增至 $2n+1$ 个，我们将二分前后两个积分值联系起来加以考察. 注意到每个子区间

$[x_k, x_{k+1}]$经过二分只增加了一个分点 $x_{k+\frac{1}{2}}=\frac{1}{2}(x_k+x_{k+1})$，用复化梯形公式求得该子区间上的积分值为

$$\frac{h}{4}[f(x_k)+2f(x_{k+\frac{1}{2}})+f(x_{k+1})].$$

注意，这里 $h=\frac{b-a}{n}$代表二分前的步长. 将每个子区间上的积分值相加得

$$T_{2n}=\frac{h}{4}\sum_{k=0}^{n-1}[f(x_k)+f(x_{k+1})]+\frac{h}{2}\sum_{k=0}^{n-1}f(x_{k+\frac{1}{2}}),$$

从而利用式(3.2)可导出下列递推公式

$$T_{2n}=\frac{1}{2}T_n+\frac{h}{2}\sum_{k=0}^{n-1}f(x_{k+\frac{1}{2}}). \tag{4.1}$$

例 2　计算积分值

$$I=\int_0^1\frac{\sin x}{x}\mathrm{d}x.$$

解　我们先对整个区间[0,1]使用梯形公式. 对于函数 $f(x)=\frac{\sin x}{x}$，它在 $x=0$ 的值定义为 $f(0)=1$，而 $f(1)=0.8414709$，据梯形公式计算得

$$T_1=\frac{1}{2}[f(0)+f(1)]=0.9207355.$$

然后将区间二等分，再求出中点的函数值

$$f\left(\frac{1}{2}\right)=0.9588510,$$

从而利用递推公式(4.1)，有

$$T_2=\frac{1}{2}T_1+\frac{1}{2}f\left(\frac{1}{2}\right)=0.9397933.$$

我们进一步二分求积区间，并计算新分点上的函数值

$$f(1/4)=0.9896158,\ f(3/4)=0.9088516.$$

再利用式(4.1),有

$$T_4=\frac{1}{2}T_2+\frac{1}{4}\left[f\left(\frac{1}{4}\right)+f\left(\frac{3}{4}\right)\right]=0.9445135.$$

这样不断二分下去,计算结果见下表(表中 k 代表二分次数,区间等分数 $n=2^k$).

表 4-3

k	1	2	3	4	5
T_n	0.9397933	0.9445135	0.9456909	0.9459850	0.9460596
k	6	7	8	9	10
T_n	0.9460769	0.9460815	0.9460827	0.9460830	0.9460831

它表明用复化梯形公式计算积分 I 要达到 7 位有效数字的精度需要二分区间 10 次,即要有分点 1025 个,计算量很大.

4.4.2 龙贝格算法

梯形法计算简单但收敛慢,如何提高收敛速度以节省计算量是本节要讨论的中心问题. 根据复化梯形公式的余项表达式(3.3)可知

$$I-T_n=-\frac{b-a}{12}h^2f''(\eta),\quad \eta\in(a,b);$$

$$I-T_{2n}=-\frac{b-a}{12}\left(\frac{h}{2}\right)^2f''(\bar{\eta}),\quad \bar{\eta}\in(a,b).$$

假定 $f''(\eta)\approx f''(\bar{\eta})$,则有

$$\frac{I-T_{2n}}{I-T_n}\approx\frac{1}{4}.$$

将上式移项整理,可得

$$I-T_{2n}\approx\frac{1}{3}(T_{2n}-T_n).\tag{4.2}$$

由此可见,只有二分前后的两个积分值 T_n 与 T_{2n} 相当接近,就可以保证计算结果 T_{2n} 的误差很小. 这样直接用计算结果来估计误差的方法通常称作误差的**事后估计法**.

按式(4.2),积分近似值 T_{2n} 的误差大致等于 $\frac{1}{3}(T_{2n}-T_n)$,因此如果用这个误差值作为 T_{2n} 的一种补偿,可以期望,所得到的

$$\overline{T}=T_{2n}+\frac{1}{3}(T_{2n}-T_n)=\frac{4}{3}T_{2n}-\frac{1}{3}T_n \tag{4.3}$$

可能是更好的结果.

再考察例 2,所求得的两个梯形值 $T_4=0.9445135$ 和 $T_8=0.9456909$ 的精度都很差(与准确值 $I=0.9460831$ 比较,只有两、三位有效数字),但如果将它们按式(4.3)做线性组合,则新的近似值

$$\overline{T}=\frac{4}{3}T_8-\frac{1}{3}T_4=0.9460833$$

却有 6 位有效数字.

按公式(4.3)组合得到的近似值 $\overline{T}$,其实质究竟是什么呢? 直接验证易知

$$S_n=\frac{4}{3}T_{2n}-\frac{1}{3}T_n. \tag{4.4}$$

这就是说,用梯形法二分前后的两个积分值 T_n 与 T_{2n},按式(4.3)做线性组合,结果得到辛普森法的积分值 S_n.

再考察辛普森法,按误差公式(3.6),其截断误差大致与 h^4 成正比,因此,若将步长折半则误差将减至原有误差的 1/16,即有

$$\frac{I-S_{2n}}{I-S_n}\approx\frac{1}{16},$$

由此得到

$$I\approx\frac{16}{15}S_{2n}-\frac{1}{15}S_n.$$

不难直接验证，上式右端的值等于 C_n，C_n 为复化柯特斯公式，它的精度为 $O(h^6)$，就是说，用辛普森法二分前后两个积分值 S_n 与 S_{2n}，可得到 S_{2n}的近似误差估计为$\frac{1}{15}(S_{2n}-S_n)$，并且按上式做线性组合可得复化柯特斯的积分值 C_n，即有

$$C_n = \frac{16}{15}S_{2n} - \frac{1}{15}S_n. \tag{4.5}$$

重复同样的手续，依据柯特斯法的误差阶为 h^6，可进一步导出下列**龙贝格**(Romberg)**公式**：

$$R_n = \frac{64}{63}C_{2n} - \frac{1}{63}C_n. \tag{4.6}$$

我们在变步长的过程中运用公式(4.4)，(4.5)和(4.6)，就能将粗糙的梯形值 T_n 逐步加工成精度较高的辛普森值 S_n、柯特斯值 C_n 和龙贝格值 R_n.

例 3 用加速公式(4.4)，(4.5)和(4.6)加工例 2 得到的梯形值，计算结果见下表(k 代表二分次数)：

表 4-4

k	T_{2^k}	$S_{2^{k-1}}$	$C_{2^{k-2}}$	$R_{2^{k-3}}$
0	0.9207355			
1	0.9397933	0.9461459		
2	0.9445135	0.9460869	0.9460830	
3	0.9456909	0.9460833	0.9460831	0.9460831

我们看到，这里利用二分 3 次的数据(它们的精度都很差，只有两三位有效数字)，通过三次加速求得 $R_1=0.9460831$，这个结果的每一位数字都是有效数字，可见加速的效果是十分显著的.

4.4.3 理查森外推加速法

上面讨论说明由梯形公式出发，将区间$[a,b]$逐次二分可提高

求积公式的精度，上述加速过程还可继续下去，其理论依据是梯形公式的余项展开，设

$$I - T_n = -\frac{b-a}{12}h^2 f''(\eta),\ \eta \in [a,b],\ h = \frac{b-a}{n}.$$

若记 $T_n = T(h)$，当区间$[a,b]$分为 $2n$ 等分时，则有 $T_{2n} = T\left(\frac{h}{2}\right)$，并且有

$$T(h) = I + \frac{b-a}{12}h^2 f''(\eta),\ \lim_{h\to 0} T(h) = T(0) = I,$$

可证明梯形公式余项可展成级数形式，即

定理 4　设 $f(x) \in C^\infty[a,b]$，则有

$$T(h) = I + \alpha_1 h^2 + \alpha_2 h^4 + \cdots + \alpha_l h^{2l} + \cdots, \tag{4.7}$$

其中系数 $\alpha_l (l=1,2,\cdots)$ 与 h 无关.

此定理可利用 $f(x)$ 的泰勒展开推导得到，此处从略.

定理 4 表明 $T(h) \approx I$ 是 $O(h^2)$ 阶，在(4.7)中，若用 $h/2$ 代替 h，有

$$T\left(\frac{h}{2}\right) = I + \alpha_1 \frac{h^2}{4} + \alpha_2 \frac{h^4}{16} + \cdots + \alpha_l \left(\frac{h}{2}\right)^{2l} + \cdots, \tag{4.8}$$

若用 4 乘(4.8)式，减去(4.7)式再除 3 记之为 $T_1(h)$，则得

$$T_1(h) = \frac{4T\left(\frac{h}{2}\right) - T(h)}{3} = I + \beta_1 h^4 + \beta_2 h^6 + \cdots, \tag{4.9}$$

这里 $\beta_1, \beta_2, \cdots$ 以及后面将出现的 r_k, δ_k 均为与 h 无关的系数，这样构造的 $T_1(h)$ 与积分值 I 近似的阶为 $O(h^4)$. 比较(4.9)与(4.4)可知，这样构造的序列 $T_1(h)$，$T_1\left(\frac{h}{2}\right)$，…就是辛普森公式序列 S_n，S_{2n}，….

又根据(4.9)有

$$T_1\left(\frac{h}{2}\right)=I+\beta_1\frac{h^4}{16}+\beta_2h^6+\beta_3h^8+\cdots,$$

若令

$$T_2(h)=\frac{16}{15}T_1\left(\frac{h}{2}\right)-\frac{1}{16}T_1(h),$$

则又可进一步从余项展开式中消去 h^4 项，而有

$$T_2(h)=I+\gamma_1h^6+\gamma_2h^8+\cdots.$$

这样构造出的 $\{T_2(h)\}$，其实就是柯特斯公式序列，它与积分值 I 的逼近阶为 $O(h^6)$. 如此继续下去，每加速一次，误差的量级便提高 2 阶，一般地，若记 $T_0(h)=T(h)$，则有

$$T_m(h)=\frac{4^m}{4^m-1}T_{m-1}\left(\frac{h}{2}\right)-\frac{1}{4^m-1}T_{m-1}(h). \tag{4.10}$$

经过 $m(m=1,2,\cdots)$ 次加速后，余项便取下列形式：

$$T_m(h)=I+\delta_1h^{2(m+1)}+\delta_2h^{2(m+2)}+\cdots. \tag{4.11}$$

上述处理方法通常称为**理查森**(Richardson)**外推加速方法**.

设以 $T_0^{(k)}$ 表示二分 k 次后求得的梯形值，且以 $T_m^{(k)}$ 表示序列 $\{T_0^{(k)}\}$ 的 m 次加速值，则依递推公式(4.10)可得

$$T_m^{(k)}=\frac{4^m}{4^m-1}T_{m-1}^{(k+1)}-\frac{1}{4^m-1}T_{m-1}^{(k)}\quad(k=1,2,\cdots). \tag{4.12}$$

公式(4.12)也称为**龙贝格求积算法**，计算过程如下：

(1) 取 $k=0$, $h=b-a$, 求 $T_0^{(0)}=\frac{h}{2}[f(a)+f(b)]$.

令 $1\to k$(k 记区间 $[a,b]$ 的二分次数).

(2) 求梯形值 $T_0\left(\frac{b-a}{2^k}\right)$，即按递推公式(4.1)计算 $T_0^{(k)}$.

(3) 求加速值，按公式(4.12)逐个求出 T 表(见表 4-5)的第 k 行其余各元素 $T_j^{(k-j)}(j=1,2,\cdots,k)$.

(4) 若 $|T_k^{(0)}-T_{k-1}^{(0)}|<\varepsilon$(预先给定的精度)，则终止计算，并取

$T_k^{(0)} \approx I$；否则令 $k+1 \to k$ 转(2)继续计算.

表 4-5 T 表

k	h	$T_0^{(k)}$		$T_1^{(k)}$		$T_2^{(k)}$		$T_3^{(k)}$		$T_4^{(k)}$	…
0	$b-a$	$T_0^{(0)}$									
1	$\frac{b-a}{2}$	$T_0^{(1)}$	↓①	$T_1^{(0)}$							
2	$\frac{b-a}{4}$	$T_0^{(2)}$	↓②	$T_1^{(1)}$	↓③	$T_2^{(0)}$					
3	$\frac{b-a}{8}$	$T_0^{(3)}$	↓④	$T_1^{(2)}$	↓⑤	$T_2^{(1)}$	↓⑥	$T_3^{(0)}$			
4	$\frac{b-a}{16}$	$T_0^{(4)}$	↓⑦	$T_1^{(3)}$	↓⑧	$T_2^{(2)}$	↓⑨	$T_3^{(1)}$	↓⑩	$T_4^{(0)}$	
⋮	⋮	⋮		⋮		⋮		⋮		⋮	⋱

表 4-5 指出了计算过程，第 2 列 $h=\frac{b-a}{2^k}$ 给出了子区间长度，ⓘ表示第 i 步外推.

可以证明，如果 $f(x)$ 充分光滑，那么 T 表每一列的元素及对角线元素均收敛到所求的积分值 I，即

$$\lim_{k\to\infty} T_m^{(k)} = I \quad (m \text{ 固定}),$$

$$\lim_{m\to\infty} T_m^{(0)} = I.$$

对于 $f(x)$ 不充分光滑的函数也可用龙贝格算法计算，只是收敛慢一些，这时也可以直接使用复化辛普森公式计算. 见下面例题.

例 4 用龙贝格算法计算积分 $I=\int_0^1 x^{3/2}\mathrm{d}x$.

解 $f(x)=x^{3/2}$ 在$[0,1]$上仅是一次连续可微，用龙贝格算法计算结果见表 4-6. 从表中看到用龙贝格算到 $k=5$ 的精度与辛普森求积精度相当. 这里 I 的精确值为 0.4.

表 4-6

k	$T_0^{(k)}$	$T_1^{(k)}$	$T_2^{(k)}$	$T_3^{(k)}$	$T_4^{(k)}$	$T_5^{(k)}$
0	0.500000					
1	0.426777	0.402369				
2	0.407018	0.400432	0.400302			
3	0.401812	0.400077	0.400054	0.400050		
4	0.400463	0.400014	0.400009	0.400009	0.400009	
5	0.400118	0.400002	0.400002	0.400002	0.400002	0.400002

4.5 高斯求积公式

4.5.1 一般理论

形如(1.3)的机械求积公式

$$\int_a^b f(x)\mathrm{d}x \approx \sum_{k=0}^{n} A_k f(x_k)$$

含有 $2n+2$ 个待定参数 $x_k, A_k(k=0,1,\cdots,n)$. 当 x_k 为等距节点时得到的插值求积公式其代数精度至少为 n 次，如果适当选取 x_k $(k=0,1,\cdots,n)$，有可能使求积公式具有 $2n+1$ 次代数精度，这类求积公式称为**高斯**(Gauss)求积公式. 为使问题更具一般性，我们研究带权积分 $I=\int_a^b f(x)\rho(x)\mathrm{d}x$，这里 $\rho(x)$ 为权函数，类似(1.3)，它的求积公式为

$$\int_a^b f(x)\rho(x)\mathrm{d}x \approx \sum_{k=0}^{n} A_k f(x_k), \tag{5.1}$$

$A_k(k=0,1,\cdots,n)$ 为不依赖于 $f(x)$ 的求积系数，$x_k(k=0,1,\cdots,n)$ 为求积节点，可适当选取 x_k 及 $A_k(k=0,1,\cdots,n)$ 使(5.1)具有 $2n+1$ 次代数精度.

定义 4 如果求积公式(5.1)具有 $2n+1$ 次代数精度,则称其节点 $x_k(k=0,1,\cdots,n)$ 为**高斯点**,相应公式(5.1)称为**高斯求积公式**.

根据定义要使(5.1)具有 $2n+1$ 次代数精度,只要取 $f(x)=x^m$,对 $m=0,1,\cdots,2n+1$,(5.1)精确成立,则得

$$\sum_{k=0}^{n} A_k x_k = \int_a^b x^m \rho(x)\mathrm{d}x \quad m=0,1,\cdots,2n+1. \tag{5.2}$$

当给定权函数 $\rho(x)$,求出右端积分,则可由(5.2)解得 A_k 及 $x_k(k=0,1,\cdots,n)$.

例 5 试构造下列积分的高斯求积公式:

$$\int_0^1 \sqrt{x} f(x)\mathrm{d}x \approx A_0 f(x_0) + A_1 f(x_1). \tag{5.3}$$

解 令公式(5.3)对于 $f(x)=1,x,x^2,x^3$ 准确成立,得

$$\begin{cases} A_0 + A_1 = \dfrac{2}{3}; \\ x_0A_0 + x_1A_1 = \dfrac{2}{5}; \\ x_0^2A_0 + x_1^2A_1 = \dfrac{2}{7}; \\ x_0^3A_0 + x_1^3A_1 = \dfrac{2}{9}. \end{cases} \tag{5.4}$$

由于

$$x_0A_0 + x_1A_1 = x_0(A_0+A_1) + (x_1-x_0)A_1,$$

利用(5.4)的第 1 式,可将第 2 式化为

$$\frac{2}{3}x_0 + (x_1-x_0)A_1 = \frac{2}{5}.$$

同样地,利用第 2 式化第 3 式,利用第 3 式化第 4 式,分别得

$$\frac{2}{5}x_0 + (x_1-x_0)x_1A_1 = \frac{2}{7};$$

$$\frac{2}{7}x_0 + (x_1 - x_0)x_1^2 A_1 = \frac{2}{9}.$$

从上面三个式子消去$(x_1-x_0)A_1$，有

$$\begin{cases} \frac{2}{5}x_0 + \left(\frac{2}{5} - \frac{2}{3}x_0\right)x_1 = \frac{2}{7}; \\ \frac{2}{7}x_0 + \left(\frac{2}{7} - \frac{2}{5}x_0\right)x_1 = \frac{2}{9}. \end{cases}$$

进一步整理得

$$\begin{cases} \frac{2}{5}(x_0 + x_1) - \frac{2}{3}x_0 x_1 = \frac{2}{7}; \\ \frac{2}{7}(x_0 + x_1) - \frac{2}{5}x_0 x_1 = \frac{2}{9}. \end{cases}$$

由此解出

$$x_0 x_1 = \frac{5}{21}, \quad x_0 + x_1 = \frac{10}{9},$$

从而求出

$$x_0 = 0.821162, \quad x_1 = 0.289949;$$

$$A_0 = 0.389111, \quad A_1 = 0.277556.$$

于是形如(5.3)的高斯公式是

$$\int_0^1 \sqrt{x} f(x) \mathrm{d}x \approx 0.389111 f(0.821162) + 0.277556 f(0.289949).$$

从此例看到求解非线性方程组(5.2)较复杂，通常 $n \geqslant 2$ 就很难求解. 故一般不通过解方程(5.2)求 x_k 及 $A_k(k=0,1,\cdots,n)$，而从分析高斯点的特性来构造高斯求积公式.

定理 5 插值型求积公式(5.1)的节点 $a \leqslant x_0 < x_1 < \cdots < x_n \leqslant b$ 是高斯点的充分必要条件是以这些节点为零点的多项式

$$\omega_{n+1}(x) = (x - x_0)(x - x_1)\cdots(x - x_n)$$

与任何次数不超过 n 的多项式 $P(x)$ 带权 $\rho(x)$ 正交，即

$$\int_a^b P(x)\omega_{n+1}(x)\rho(x)\mathrm{d}x = 0. \tag{5.5}$$

证明　必要性. 设 $P(x)\in H_n$，则 $P(x)\omega_{n+1}(x)\in H_{2n+1}$，因此，如果 $x_0,x_1,\cdots,x_n$ 是高斯点，则求积公式(5.1)对于 $f(x)=P(x)\omega_{n+1}(x)$ 精确成立，即有

$$\int_a^b P(x)\omega_{n+1}(x)\rho(x)\mathrm{d}x = \sum_{k=0}^{n} A_k P(x_k)\omega_{n+1}(x_k).$$

因 $\omega_{n+1}(x_k)=0(k=0,1,\cdots,n)$，故(5.5)成立.

再证充分性. 对于 $\forall f(x)\in H_{2n+1}$，用 $\omega_{n+1}(x)$ 除 $f(x)$，记商为 $P(x)$，余式为 $q(x)$，即 $f(x)=P(x)\omega_{n+1}(x)+q(x)$，其中 $P(x)$，$q(x)\in H_n$. 由(5.5)可得

$$\int_a^b f(x)\rho(x)\mathrm{d}x = \int_a^b q(x)\rho(x)\mathrm{d}x. \tag{5.6}$$

由于所给求积公式(5.1)是插值型的，它对于 $q(x)\in H_n$ 是精确的，即

$$\int_a^b q(x)\rho(x)\mathrm{d}x = \sum_{k=0}^{n} A_k q(x_k).$$

再注意到 $\omega_{n+1}(x_k)=0(k=0,1,\cdots,n)$，知 $q(x_k)=f(x_k)(k=0,1,\cdots,n)$，从而由(5.6)有

$$\int_a^b f(x)\rho(x)\mathrm{d}x = \int_a^b q(x)\rho(x)\mathrm{d}x = \sum_{k=0}^{n} A_k f(x_k).$$

可见求积公式(5.1)对一切次数不超过 $2n+1$ 的多项式均精确成立. 因此，$x_k(k=0,1,\cdots,n)$为高斯点. 证毕.

定理表明在$[a,b]$上带权 $\rho(x)$的 $n+1$ 次正交多项式的零点就是求积公式(5.1)的高斯点，有了求积节点 $x_k(k=0,1,\cdots,n)$，再利用(5.2)对 $m=0,1,\cdots,n$ 成立，则得到一组关于求积系数 $A_0,A_1,\cdots,A_n$ 的线性方程. 解此方程则得 $A_k(k=0,1,\cdots,n)$. 也可直接由 $x_0,x_1,\cdots,x_n$ 的插值多项式求出求积系数 A_k ($k=$

$0,1,\cdots,n$).

下面讨论高斯求积公式(5.1)的余项. 利用 $f(x)$ 在节点 $x_k(k=0,1,\cdots,n)$ 的埃尔米特插值 $H_{2n+1}(x)$,即

$$H_{2n+1}(x_k)=f(x_k),\quad H'_{2n+1}(x_k)=f'(x_k),\ k=0,1,\cdots,n.$$

于是

$$f(x)=H_{2n+1}(x)+\frac{f^{(2n+2)}(\xi)}{(2n+2)!}\omega_{n+1}^2(x)$$

两端乘 $\rho(x)$,并由 a 到 b 积分,则得

$$I=\int_a^b f(x)\rho(x)\mathrm{d}x=\int_a^b H_{2n+1}(x)\rho(x)\mathrm{d}x+R_n[f]. \quad (5.7)$$

其中右端第一项积分对 $2n+1$ 次多项式精确成立,故

$$R_n[f]=I-\sum_{k=0}^{n}A_kf(x_k)=\int_a^b\frac{f^{(2n+2)}(\xi)}{(2n+2)!}\omega_{n+1}^2(x)\rho(x)\mathrm{d}x.$$

由于 $\omega_{n+1}^2(x)\rho(x)\geqslant0$, 故由积分中值定理得(5.1)的余项为

$$R_n[f]=\frac{f^{(2n+2)}(\eta)}{(2n+2)!}\int_a^b\omega_{n+1}^2(x)\rho(x)\mathrm{d}x. \quad (5.8)$$

下面讨论高斯求积公式的稳定性与收敛性.

定理 6 高斯求积公式(5.1)的求积系数 $A_k(k=0,1,\cdots,n)$ 全是正的.

证明 考察

$$l_k(x)=\prod_{\substack{j=0\\j\neq k}}^{n}\frac{x-x_j}{x_k-x_j},$$

它是 n 次多项式,因而 $l_k^2(x)$ 是 $2n$ 次多项式,故高斯求积公式(5.1)对于它能准确成立,即有

$$0<\int_a^b l_k^2(x)\rho(x)\mathrm{d}x=\sum_{i=0}^{n}A_il_k^2(x_i).$$

注意到 $l_k(x_i)=\delta_{ki}$, 上式右端实际上即等于 A_k,从而有

$$A_k=\int_a^b l_k^2(x)\rho(x)\mathrm{d}x>0.$$

定理得证.

由本定理及定理 2,则得

推论 高斯求积公式(5.1)是稳定的.

定理 7 设 $f(x) \in C[a,b]$,则高斯求积公式(5.1)是收敛的,即

$$\lim_{n\to\infty}\sum_{k=0}^{n} A_k f(x_k) = \int_a^b f(x)\rho(x)\mathrm{d}x.$$

证明见[1].

4.5.2 高斯-勒让德求积公式

在高斯求积公式(5.1)中,若取权函数 $\rho(x)=1$,区间为$[-1,1]$,则得公式

$$\int_{-1}^{1} f(x)\mathrm{d}x \approx \sum_{k=0}^{n} A_k f(x_k). \tag{5.9}$$

我们知道勒让德多项式(见第 3 章(2.5)式)是区间$[-1,1]$上的正交多项式,因此,勒让德多项式 $P_{n+1}(x)$ 的零点就是求积公式(5.9)的高斯点. 形如(5.9)的高斯公式特别地称为**高斯-勒让德求积公式**.

若取 $P_1(x)=x$ 的零点 $x_0=0$ 做节点构造求积公式

$$\int_{-1}^{1} f(x)\mathrm{d}x \approx A_0 f(0).$$

令它对 $f(x)=1$ 准确成立,即可定出 $A_0=2$. 这样构造出的一点高斯-勒让德求积公式是中矩形公式.

再取 $P_2(x)=\frac{1}{2}(3x^2-1)$ 的两个零点 $\pm\frac{1}{\sqrt{3}}$ 构造求积公式

$$\int_{-1}^{1} f(x)\mathrm{d}x \approx A_0 f\left(-\frac{1}{\sqrt{3}}\right)+A_1 f\left(\frac{1}{\sqrt{3}}\right),$$

令它对 $f(x)=1,x$ 都准确成立,有

$$\begin{cases} A_0 + A_1 = 2; \\ A_0\left(-\frac{1}{\sqrt{3}}\right) + A_1\left(\frac{1}{\sqrt{3}}\right) = 0. \end{cases}$$

由此解出 $A_0 = A_1 = 1$，从而得到两点高斯-勒让德求积公式

$$\int_{-1}^{1} f(x)\mathrm{d}x \approx f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right).$$

三点高斯-勒让德公式的形式是

$$\int_{-1}^{1} f(x)\mathrm{d}x \approx \frac{5}{9}f\left(-\frac{\sqrt{15}}{5}\right) + \frac{8}{9}f(0) + \frac{5}{9}f\left(\frac{\sqrt{15}}{5}\right).$$

表 4-7 列出高斯-勒让德求积公式(5.9)的节点和系数.

表 4-7

n	x_k	A_k
0	0.0000000	2.0000000
1	±0.5773503	1.0000000
2	±0.7745967	0.5555556
	0.0000000	0.8888889
3	±0.8611363	0.3478548
	±0.3399810	0.6521452
4	±0.9061798	0.2369269
	±0.5384693	0.4786287
	0.0000000	0.5688889

公式(5.9)的余项由(5.8)得

$$R_n[f] = \frac{f^{(2n+2)}(\eta)}{(2n+2)!}\int_{-1}^{1} \widetilde{\mathrm{P}}_{n+1}^2(x)\mathrm{d}x \qquad \eta \in [-1,1],$$

这里 $\widetilde{\mathrm{P}}_{n+1}(x)$是最高项系数为 1 的勒让德多项式，由第 3 章(2.6)及(2.7)得

$$R_n[f] = \frac{2^{2n+3}[(n+1)!]^4}{(2n+3)[(2n+2)!]^3} f^{(2n-2)}(\eta), \qquad \eta \in (-1,1). \tag{5.10}$$

当 $n=1$ 时，有

$$R_1[f] = \frac{1}{135} f^{(4)}(\eta).$$

它比辛普森公式余项 $R_1[f] = -\frac{1}{90} f^{(4)}(\eta)$（区间为[−1,1]）还小，且比辛普森公式少算一个函数值.

当积分区间不是[−1,1]，而是一般的区间 $[a,b]$ 时，只要做变换

$$x = \frac{b-a}{2} t + \frac{a+b}{2},$$

可将 $[a,b]$ 化为[−1,1]，这时

$$\int_a^b f(x)\mathrm{d}x = \frac{b-a}{2} \int_{-1}^{1} f\left(\frac{b-a}{2} t + \frac{a+b}{2}\right)\mathrm{d}t. \tag{5.11}$$

对等式右端的积分即可使用高斯-勒让德求积公式.

例 6　用 4 点（$n=3$）的高斯-勒让德求积公式计算

$$I = \int_0^{\frac{\pi}{2}} x^2 \cos x \mathrm{d}x.$$

解　先将区间 $\left[0, \frac{\pi}{2}\right]$ 化为[−1,1]，由(5.11)有

$$I = \int_{-1}^{1} \left(\frac{\pi}{4}\right)^3 (1+t)^2 \cos \frac{\pi}{4}(1+t)\mathrm{d}t.$$

根据表 4-7 中 $n=3$ 的节点及系数值可求得

$$I \approx \sum_{k=0}^{3} A_k f(x_k) \approx 0.467402 \qquad (\text{准确值 } I = 0.467401\cdots).$$

4.5.3　高斯-切比雪夫求积公式

若 $a=-1, b=1$，且取权函数

$$\rho(x)=\frac{1}{\sqrt{1-x^2}},$$

则所建立的高斯公式为

$$\int_{-1}^{1}\frac{f(x)}{\sqrt{1-x^2}}\mathrm{d}x\approx\sum_{k=0}^{n}A_kf(x_k). \tag{5.12}$$

特别地称为**高斯-切比雪夫求积公式**. 由于区间$[-1,1]$上关于权函数$\frac{1}{\sqrt{1-x^2}}$的正交多项式是切比雪夫多项式(见第 3 章第 2 节),因此求积公式(5.12)的高斯点是 $n+1$ 次切比雪夫多项式的零点,即为

$$x_k=\cos\left(\frac{2k+1}{2n+2}\pi\right)\qquad(k=0,1,\cdots,n).$$

通过计算(见[2])可知(5.12)的系数 $A_k=\frac{\pi}{n+1}$,使用时将 $n+1$ 个节点公式改为 n 个节点,于是高斯-切比雪夫求积公式写成

$$\int_{-1}^{1}\frac{f(x)}{\sqrt{1-x^2}}\mathrm{d}x\approx\frac{\pi}{n}\sum_{k=1}^{n}f(x_k),\ x_k=\cos\frac{(2k-1)}{2n}\pi, \tag{5.13}$$

公式余项由(5.9)可算得

$$R[f]=\frac{2\pi}{2^{2n}(2n)!}f^{(2n)}(\eta),\qquad\eta\in(-1,1). \tag{5.14}$$

带权的高斯求积公式可用于计算奇异积分.

例 7 用 5 点($n=5$)的高斯-切比雪夫求积公式计算积分

$$I=\int_{-1}^{1}\frac{e^x}{\sqrt{1-x^2}}\mathrm{d}x.$$

解 这里 $f(x)=e^x$, $f^{(2n)}(x)=e^x$, 当 $n=5$ 时由公式(5.13)可得

$$I=\frac{\pi}{5}\sum_{k=1}^{5}e^{\cos\frac{2k-1}{10}\pi}=3.977463.$$

由余项(5.14)可估计误差

$$|R[f]| \leqslant \frac{\pi}{2^9 \cdot 10!} e \leqslant 4.6 \times 10^{-9}.$$

4.6 数值微分

4.6.1 中点方法与误差分析

数值微分就是用函数值的线性组合近似函数在某点的导数值. 按导数定义可以简单地用差商近似导数,这样立即得到几种数值微分公式

$$f'(a) \approx \frac{f(a+h)-f(a)}{h},$$

$$f'(a) \approx \frac{f(a)-f(a-h)}{h},$$

$$f'(a) \approx \frac{f(a+h)-f(a-h)}{2h}. \tag{6.1}$$

其中 h 为一增量,称为**步长**,后一种数值微分方法称为中点方法,它其实是前两种方法的算术平均. 但它的误差阶却由 $O(h)$ 提高到 $O(h^2)$. 上面给出的三个公式是很实用的. 尤其是中点公式更为常用.

为要利用中点公式

$$G(h) = \frac{f(a+h)-f(a-h)}{2h}$$

计算导数 $f'(a)$ 的近似值,首先必须选取合适的步长,为此需要进行误差分析. 分别将 $f(a \pm h)$ 在 $x=a$ 处做泰勒展开有

$$f(a \pm h) = f(a) \pm hf'(a) + \frac{h^2}{2!}f''(a) \pm \frac{h^3}{3!}f'''(a)$$

$$+ \frac{h^4}{4!}f^{(4)}(a) \pm \frac{h^5}{5!}f^{(5)}(a) + \cdots$$

代入上式得

$$G(h)=f'(a)+\frac{h^2}{3!}f'''(a)+\frac{h^4}{5!}f^{(5)}(a)+\cdots$$

由此得知，从截断误差的角度看，步长越小，计算结果越准确．且

$$|f'(a)-G(h)|\leqslant\frac{h^2}{6}M,\qquad(6.2)$$

其中 $M\geqslant\max\limits_{|x-a|\leqslant h}|f'''(x)|$．

再考察舍入误差．按中点公式计算，当 h 很小时，因 $f(a+h)$ 与 $f(a-h)$ 很接近，直接相减会造成有效数字的严重损失（参看第 1 章第 4 节）．因此，从舍入误差的角度来看，步长是不宜太小的．

例如，用中点公式求 $f(x)=\sqrt{x}$ 在 $x=2$ 处的一阶导数

$$G(h)=\frac{\sqrt{2+h}-\sqrt{2-h}}{2h}.$$

设取 4 位数字计算．结果见表 4-8（导数的准确值 $f'(2)=0.353553$）．

表 4-8

h	$G(h)$	h	$G(h)$	h	$G(h)$
1	0.3660	0.05	0.3530	0.001	0.3500
0.5	0.3564	0.01	0.3500	0.0005	0.3000
0.1	0.3535	0.005	0.3500	0.0001	0.3000

从表 4-8 中看到 $h=0.1$ 的逼近效果最好，如果进一步缩小步长，则逼近效果反而越差．这里因为当 $f(a+h)$ 及 $f(a-h)$ 分别有差入误差 ε_1 及 ε_2．若令 $\varepsilon=\max\{|\varepsilon_1|,|\varepsilon_2|\}$，则计算 $f'(a)$ 的舍入误差上界为

$$\delta(f'(a))=|f'(a)-G(a)|\leqslant\frac{|\varepsilon_1|+|\varepsilon_2|}{2h}=\frac{\varepsilon}{h},$$

它表明 h 越小，舍入误差 $\delta(f'(a))$ 越大，故它是病态的. 用中点公式(6.1)计算 $f'(a)$ 的误差上界为

$$E(h)=\frac{h^2}{6}M+\frac{\varepsilon}{h},$$

要使误差 $E(h)$ 最小，步长 h 不宜太大，也不宜太小. 其最优步长应为 $h_{\text{opt}}=\sqrt[3]{3\varepsilon/M}$.

4.6.2 插值型的求导公式

对于列表函数 $y=f(x)$：

x	x_0	x_1	x_2	…	x_n
y	y_0	y_1	y_2	…	y_n

运用插值原理，可以建立插值多项式 $y=P_n(x)$ 作为它的近似. 由于多项式的求导比较容易，我们取 $P_n'(x)$ 的值作为 $f'(x)$ 的近似值，这样建立的数值公式

$$f'(x)\approx P_n'(x) \tag{6.3}$$

统称**插值型的求导公式**.

必须指出，即使 $f(x)$ 与 $P_n(x)$ 的值相差不多，导数的近似值 $P_n'(x)$ 与导数的真值 $f'(x)$ 仍然可能差别很大，因而在使用求导公式(6.3)时应特别注意误差的分析.

依据插值余项定理，求导公式(6.3)的余项为

$$f'(x)-P_n'(x)=\frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}'(x)+\frac{\omega_{n+1}(x)}{(n+1)!}\frac{\mathrm{d}}{\mathrm{d}x}f^{(n+1)}(\xi),$$

式中

$$\omega_{n+1}(x)=\prod_{i=0}^{n}(x-x_i).$$

在这一余项公式中，由于 ξ 是 x 的未知函数，我们无法对它的第二项 $\frac{\omega_{n+1}(x)}{(n+1)!}\frac{\mathrm{d}}{\mathrm{d}x}f^{(n+1)}(\xi)$ 做出进一步的说明. 因此，对于随意给出的点 x，误差 $f'(x)-P'_n(x)$ 是无法预估的. 但是，如果我们限定求某个节点 x_k 上的导数值，那么上面的第二项因式 $\omega_{n+1}(x_k)$ 变为零，这时有余项公式

$$f'(x_k)-P'_n(x_k)=\frac{f^{(n+1)}(\xi)}{(n+1)!}\omega'_{n+1}(x_k). \tag{6.4}$$

下面我们仅仅考察节点处的导数值. 为简化讨论，假定所给的节点是等距的.

1. 两点公式

设已给出两个节点 x_0,x_1 上的函数值 $f(x_0),f(x_1)$，做线性插值得公式

$$P_1(x)=\frac{x-x_1}{x_0-x_1}f(x_0)+\frac{x-x_0}{x_1-x_0}f(x_1).$$

对上式两端求导，记 $x_1-x_0=h$，有

$$P'_1(x)=\frac{1}{h}[-f(x_0)+f(x_1)],$$

于是有下列求导公式：

$$P'_1(x_0)=\frac{1}{h}[f(x_1)-f(x_0)];$$

$$P'_1(x_1)=\frac{1}{h}[f(x_1)-f(x_0)].$$

而利用余项公式(6.4)知，带余项的两点公式是

$$f'(x_0)=\frac{1}{h}[f(x_1)-f(x_0)]-\frac{h}{2}f''(\xi);$$

$$f'(x_1)=\frac{1}{h}[f(x_1)-f(x_0)]+\frac{h}{2}f''(\xi).$$

2. 三点公式

设已给出三个节点 $x_0,x_1=x_0+h,\ x_2=x_0+2h$ 上的函数值，

做二次插值

$$P_2(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}f(x_0) + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}f(x_1) + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}f(x_2),$$

令 $x=x_0+th$，上式可表示为

$$P_2(x_0+th) = \frac{1}{2}(t-1)(t-2)f(x_0) - t(t-2)f(x_1) + \frac{1}{2}t(t-1)f(x_2).$$

两端对 t 求导，有

$$P_2'(x_0+th) = \frac{1}{2h}[(2t-3)f(x_0) - (4t-4)f(x_1) + (2t-1)f(x_2)]. \tag{6.5}$$

这里撇号(′)表示对变量 x 求导数. 上式分别取 $t=0,1,2$，得到三种三点公式：

$$P_2'(x_0) = \frac{1}{2h}[-3f(x_0)+4f(x_1)-f(x_2)];$$

$$P_2'(x_1) = \frac{1}{2h}[-f(x_0)+f(x_2)];$$

$$P_2'(x_2) = \frac{1}{2h}[f(x_0)-4f(x_1)+3f(x_2)].$$

而带余项的三点求导公式如下：

$$\begin{aligned} f'(x_0) &= \frac{1}{2h}[-3f(x_0)+4f(x_1)-f(x_2)] + \frac{h^2}{3}f'''(\xi); \\ f'(x_1) &= \frac{1}{2h}[-f(x_0)+f(x_2)] - \frac{h^2}{6}f'''(\xi); \\ f'(x_2) &= \frac{1}{2h}[f(x_0)-4f(x_1)+3f(x_2)] + \frac{h^2}{3}f'''(\xi). \end{aligned} \tag{6.6}$$

其中的公式(6.6)是我们所熟悉的中点公式. 在三点公式中,它由于少用了一个函数值 $f(x_1)$而引人注目.

用插值多项式 $P_n(x)$作为 $f(x)$的近似函数,还可以建立高阶数值微分公式:

$$f^{(k)}(x) \approx P_m^{(k)}(x), \quad k = 1,2,\cdots$$

例如,将式(6.5)再对 t 求导一次,有

$$P_2''(x_0 + th) = \frac{1}{h^2}[f(x_0) - 2f(x_1) + f(x_2)],$$

于是有

$$P_2''(x_1) = \frac{1}{h^2}[f(x_1 - h) - 2f(x_1) + f(x_1 + h)].$$

而带余项的二阶三点公式如下:

$$f''(x_1) = \frac{1}{h^2}[f(x_1 - h) - 2f(x_1) + f(x_1 + h)] - \frac{h^2}{12}f^{(4)}(\xi). \tag{6.7}$$

4.6.3 利用数值积分求导

微分是积分的逆运算,因此可利用数值积分的方法来计算数值微分. 设 $f(x)$是一个充分光滑的函数,设 $\varphi(x)=f'(x)$, $x_k=a+kh$, $k=0,1,\cdots,n$, $h=\dfrac{b-a}{n}$,则有

$$f(x_{k+1}) = f(x_{k-1}) + \int_{x_{k-1}}^{x_{k+1}} \varphi(x)\mathrm{d}x \quad (k = 1,\cdots,n-1), \tag{6.8}$$

对上式右边积分采用不同的求积公式就可得到不同的数值微分公式. 例如,对 $\int_{x_{k-1}}^{x_{k+1}} \varphi(x)\mathrm{d}x$ 用中矩形公式(1.2),则得

$$\int_{x_{k-1}}^{x_{k+1}} \varphi(x)\mathrm{d}x = 2h\varphi(x_k) + \frac{1}{24}(2h)^3\varphi''(\xi_k), \quad \xi_k \in (x_{k-1}, x_{k+1}).$$

从而得到中点微分公式

$$f'(x_k)=\frac{f(x_{k+1})-f(x_{k-1})}{2h}-\frac{1}{6}h^2 f'''(\xi_k).$$

若对(6.8)右端积分用辛普森求积公式,则有

$$\int_{x_{k-1}}^{x_{k+1}}\varphi(x)\mathrm{d}x=\frac{h}{3}[\varphi(x_{k-1})+4\varphi(x_k)+\varphi(x_{k+1})]$$

$$-\frac{h^5}{90}\varphi^{(4)}(\eta_k),\qquad \eta_k\in(x_{k-1},x_{k+1}),$$

上式略去余项,并记 $\varphi(x_k)=f'(x_k)$ 的近似值为 m_k,则得到辛普森数值微分公式

$$m_{k-1}+4m_k+m_{k+1}=\frac{3}{h}[f(x_{k+1})-f(x_{k-1})]\quad(k=1,\cdots,n-1).$$

这是关于 $m_0,m_1,\cdots,m_n$ 这 $n+1$ 个未知量的 $n-1$ 个方程组,若 $m_0=f'(x_0),m_n=f'(x_n)$ 已知,则可得

$$\begin{pmatrix}4&1&&&\\1&4&1&&\\&\ddots&\ddots&\ddots&\\&&1&4&1\\&&&1&4\end{pmatrix}\begin{pmatrix}m_1\\m_2\\\vdots\\m_{n-2}\\m_{n-1}\end{pmatrix}=\begin{pmatrix}\frac{3}{h}[f(x_2)-f(x_0)]-f'(x_0)\\\frac{3}{h}[f(x_3)-f(x_1)]\\\vdots\\\frac{3}{h}[f(x_{n-1})-f(x_{n-3})]\\\frac{3}{h}[f(x_n)-f(x_{n-2})]-f'(x_n)\end{pmatrix}\tag{6.9}$$

这是关于 $m_1,\cdots,m_{n-1}$ 的三对角方程组,且系数矩阵为严格对角占优的,可用追赶法求解(见第 5 章 5.4 节).

如果端点导数值不知道,那么对(6.9)中第 1 个和第 $n-1$ 个方程可分别用 $f'(x_1)$ 及 $f'(x_{n-1})$ 的中点微分公式近似,即取

$$m_1=\frac{1}{2h}[f(x_2)-f(x_0)],\quad m_{n-1}=\frac{1}{2h}[f(x_n)-f(x_0)].$$

然后求 $m_2,\cdots,m_{n-2}$ 即为 $f'(x_2),\cdots,f'(x_{n-2})$ 的近似值.

例 8 给定 $f(x)=\sqrt{x}$ 的一张数据表(表 4-9 左部),并给定 $f'(100)$ 及 $f'(105)$ 的值(见表 4-9). 利用辛普森数值微分公式求 $f(x)$ 在 $x=101,102,103,104$ 上的一阶导数.

解 根据(6.9)有

$$\begin{pmatrix}4 & 1 & & \\ 1 & 4 & 1 & \\ & 1 & 4 & 1 \\ & & 1 & 4\end{pmatrix}\begin{pmatrix}m_1 \\ m_2 \\ m_3 \\ m_4\end{pmatrix}=\begin{pmatrix}0.24851482 \\ 0.29704785 \\ 0.29560227 \\ 0.24538260\end{pmatrix}.$$

解之得 $m_i(i=1,2,3,4)$,结果见表 4-9.

表 4-9

k	x_k	$f(x_k)=\sqrt{x_k}$	$f'(x_k)$	$m_k\approx f'(x_k)$
0	100	10.00000000	0.05000000	
1	101	10.04987562	0.049751859	0.04975186
2	102	10.09950494	0.049507377	0.049507376
3	103	10.14889157	0.049266463	0.049266463
4	104	10.19803903	0.049029033	0.049029033
5	105	10.24695077	0.048795003	

4.6.4 三次样条求导

三次样条函数 $S(x)$ 作为 $f(x)$ 的近似,不但函数值很接近,导数值也很接近,并有

$$\|f^{(k)}(x)-S^{(k)}(x)\|_\infty\leqslant C_k\|f^{(4)}\|_\infty h^{4-k}\qquad(k=0,1,2).\tag{6.10}$$

(见第 2 章定理 4),因此利用三次样条函数 $S(x)$ 直接得到

$$f^{(k)}(x)\approx S^{(k)}(x)\qquad(k=0,1,2).$$

根据第 2 章(7.8),(7.9)可求得

$$f'(x_k)\approx S'(x_k)=-\frac{h_k}{3}M_k-\frac{h_k}{6}M_{k+1}+f[x_k,x_{k+1}],$$

$$f''(x_k) = M_k.$$

这里 $f[x_k, x_{k-1}]$为一阶均差. 其误差由(6.10)可得

$$\| f' - S' \|_\infty \leqslant \frac{1}{24} \| f^{(4)} \|_\infty h^3,$$

$$\| f'' - S'' \|_\infty \leqslant \frac{3}{8} \| f^{(4)} \|_\infty h^2.$$

4.6.5 数值微分的外推算法

利用中点公式计算导数值时

$$f'(x) \approx G(h) = \frac{1}{2h}[f(x+h) - f(x-h)].$$

对 $f(x)$在点 x 做泰勒级数展开有

$$f'(x) = G(h) + \alpha_1 h^2 + \alpha_2 h^4 + \cdots,$$

其中 $\alpha_i (i=1,2,\cdots)$与 h 无关,利用理查森外推(见本章第 4 节)对 h 逐次分半,若记 $G_0(h)=G(h)$,则有

$$G_m(h) = \frac{4^m G_{m-1}\left(\frac{h}{2}\right) - G_{m-1}(h)}{4^m - 1} \quad (m = 1,2,\cdots). \tag{6.11}$$

公式(6.11)的计算过程见表 4-10,表中ⓘ为外推步数.

表 4-10

$G(h)$				
$G\left(\frac{h}{2}\right)$ ↓①	$G_1(h)$			
$G\left(\frac{h}{2^2}\right)$ ↓②	$G_1\left(\frac{h}{2}\right)$ ↓③	$G_2(h)$		
$G\left(\frac{h}{2^3}\right)$ ↓④	$G_1\left(\frac{h}{2^2}\right)$ ↓⑤	$G_2\left(\frac{h}{2}\right)$ ↓⑥	$G_3(h)$	
⋮	⋮	⋮	⋮	⋱

根据理查森外推方法,(6.11)的误差为

$$f'(x) - G_m(h) = O(h^{2(m+1)}).$$

由此看出当 m 较大时,计算是很精确的. 考虑到舍人误差,一般 m 不能取太大.

例 9　用外推法计算 $f(x)=x^2e^{-x}$在 $x=0.5$ 的导数.

解　令 $G(h)=\frac{1}{2h}\left[\left(\frac{1}{2}+h\right)^2 e^{-\left(\frac{1}{2}+h\right)}-\left(\frac{1}{2}-h\right)^2 e^{-\left(\frac{1}{2}-h\right)}\right]$,

当 $h=0.1, 0.05, 0.025$ 时,由外推法表 4-10 可算得

$G(0.1) = 0.4516049081$

$G(0.05) = 0.4540761693$ ↓①→ $G_1(h) = 0.4548999231$

$G(0.025) = 0.4546926288$ ↓②→ $G_1\left(\frac{h}{2}\right) = 0.4548981152$

↓③→ $G_2 = 0.454897994$

$f'(0.5)$的精确值为 0.454897994,可见当 $h=0.025$ 时用中点微分公式只有 3 位有效数字,外推一次达到 5 位有效数字,外推两次达到 9 位有效数字.

评　注

本章介绍积分和微分的数值计算方法. 我们知道,积分和微分是两种分析运算,它们都是用极限来定义的. 数值积分和数值微分则归结为函数值的四则运算,从而使计算过程可以在计算机上完成.

处理数值积分和数值微分的基本方法是逼近法:设法构造某个简单函数 $P(x)$近似 $f(x)$,然后对 $P(x)$求积(求导)得到 $f(x)$的积分(导数)的近似值. 本章基于插值原理推导了数值积分和数值微分的基本公式.

插值求积公式分牛顿-柯特斯公式和高斯公式两类. 前者为

等距节点，但 $n \geqslant 8$ 时计算不稳定，实际计算宜采用复合求积方法. 高斯公式精度高，计算稳定，但节点选取较困难. 带权高斯求积方法，能把复杂积分化简，还可以直接计算奇异积分. 这两类公式也可通过求积公式的代数精度建立.

基于理查森外推的龙贝格求积方法由于计算程序简单，精度较高，是一个在计算机上求积的有效算法. 在数值微分中也有相似的算法，外推方法和思想是数值分析中一种很重要的方法.

数值微分由于计算不稳定性，步长选取是很重要的. 通常数值微分的外推法可得到较满意结果，但 h 也不能太小.

数值积分中一些重要内容如奇异积分，振荡函数积分和二重积分计算等均未涉及，可参见文献[1].

习 题

1. 确定下列求积公式中的待定参数，使其代数精度尽量高，并指明所构造出的求积公式所具有的代数精度：

1) $\int_{-h}^{h} f(x)\mathrm{d}x \approx A_{-1} f(-h) + A_0 f(0) + A_1 f(h)$；

2) $\int_{-2h}^{2h} f(x)\mathrm{d}x \approx A_{-1} f(-h) + A_0 f(0) + A_1 f(h)$；

3) $\int_{-1}^{1} f(x)\mathrm{d}x \approx [f(-1) + 2f(x_1) + 3f(x_2)]/3$；

4) $\int_{0}^{h} f(x)\mathrm{d}x \approx h[f(0) + f(h)]/2 + ah^2[f'(0) - f'(h)]$.

2. 分别用梯形公式和辛普森公式计算下列积分：

1) $\int_{0}^{1} \frac{x}{4+x^2}\mathrm{d}x, \quad n = 8$；

2) $\int_{0}^{1} \frac{(1-\mathrm{e}^{-x})^{\frac{1}{2}}}{x}\mathrm{d}x, \quad n = 10$；

3) $\int_{1}^{9} \sqrt{x}\mathrm{d}x, \quad n = 4$；

4) $\int_0^{\pi/6}\sqrt{4-\sin^2\varphi}\mathrm{d}\varphi$, $n=6$.

3. 直接验证柯特斯公式(2.4)具有5次代数精度.

4. 用辛普森公式求积分 $\int_0^1 \mathrm{e}^{-x}\mathrm{d}x$ 并估计误差.

5. 推导下列三种矩形求积公式：

$$\int_a^b f(x)\mathrm{d}x=(b-a)f(a)+\frac{f'(\eta)}{2}(b-a)^2;$$

$$\int_a^b f(x)\mathrm{d}x=(b-a)f(b)-\frac{f'(\eta)}{2}(b-a)^2;$$

$$\int_a^b f(x)\mathrm{d}x=(b-a)f\left(\frac{a+b}{2}\right)+\frac{f''(\eta)}{24}(b-a)^3.$$

6. 若用复化梯形公式计算积分 $I=\int_0^1 \mathrm{e}^x\mathrm{d}x$，问区间[0,1]应分多少等分才能使截断误差不超过 $\frac{1}{2}\times10^{-5}$？若改用复化辛普森公式，要达到同样精度区间[0,1]应分多少等分？

7. 如果 $f''(x)>0$，证明用梯形公式计算积分 $I=\int_a^b f(x)\mathrm{d}x$ 所得结果比准确值 I 大，并说明其几何意义.

8. 用龙贝格求积方法计算下列积分，使误差不超过 10^{-5}.

(1) $\frac{2}{\sqrt{\pi}}\int_0^1 \mathrm{e}^{-x}\mathrm{d}x$,

(2) $\int_0^{2\pi} x\sin x\mathrm{d}x$,

(3) $\int_0^3 x\sqrt{1+x^2}\mathrm{d}x$.

9. 用 $n=2,3$ 的高斯-勒让德公式计算积分

$$\int_1^3 \mathrm{e}^x\sin x\mathrm{d}x..$$

10. 地球卫星轨道是一个椭圆，椭圆周长的计算公式是

$$S=a\int_0^{\pi/2}\sqrt{1-\left(\frac{c}{a}\right)^2\sin^2\theta}\ \mathrm{d}\theta,$$

这里 a 是椭圆的半长轴，c 是地球中心与轨道中心(椭圆中心)的距离，记 h 为近地点距离，H 为远地点距离，$R=6371$(km)为地球半径，则

$$a=(2R+H+h)/2,\ c=(H-h)/2.$$

我国第一颗人造地球卫星近地点距离 $h=439(\mathrm{km})$，远地点距离 $H=2384(\mathrm{km})$，试求卫星轨道的周长.

11. 证明等式

$$n\sin\frac{\pi}{n}=\pi-\frac{\pi^3}{3!n^2}+\frac{\pi^5}{5!n^4}-\cdots$$

试依据 $n\sin(\pi/n)(n=3,6,12)$ 的值，用外推算法求 π 的近似值.

12. 用下列方法计算积分 $\int_1^3\frac{\mathrm{d}y}{y}$，并比较结果.

1) 龙贝格方法；

2) 三点及五点高斯公式；

3) 将积分区间分为四等分，用复化两点高斯公式.

13. 用三点公式和积分方法求 $f(x)=\frac{1}{(1+x)^2}$ 在 $x=1.0,1.1$ 和 1.2 处的导数值，并估计误差. $f(x)$ 的值由下表给出：

x	1.0	1.1	1.2
$f(x)$	0.2500	0.2268	0.2066

第5章　解线性方程组的直接方法

5.1　引言与预备知识

5.1.1　引言

在自然科学和工程技术中很多问题的解决常常归结为解线性代数方程组，例如电学中的网络问题，船体数学放样中建立三次样条函数问题，用最小二乘法求实验数据的曲线拟合问题，解非线性方程组问题，用差分法或者有限元方法解常微分方程、偏微分方程边值问题等都导致求解线性代数方程组，而这些方程组的系数矩阵大致分为两种，一种是低阶稠密矩阵(例如，阶数不超过150)，另一种是大型稀疏矩阵(即矩阵阶数高且零元素较多).

关于线性方程组的数值解法一般有两类：

1. 直接法

就是经过有限步算术运算，可求得方程组精确解的方法(若计算过程中没有舍入误差). 但实际计算中由于舍入误差的存在和影响，这种方法也只能求得线性方程组的近似解. 本章将阐述这类算法中最基本的高斯消去法及其某些变形. 这类方法是解低阶稠密矩阵方程组及某些大型稀疏方程组(例如，大型带状方程组)的有效方法.

2. 迭代法

就是用某种极限过程去逐步逼近线性方程组精确解的方法. 迭代法具有需要计算机的存贮单元较少、程序设计简单、原始系数矩阵在计算过程中始终不变等优点，但存在收敛性及收敛速度问

题. 迭代法是解大型稀疏矩阵方程组(尤其是由微分方程离散后得到的大型方程组)的重要方法(见第 6 章).

为了讨论线性方程组数值解法,需复习一些基本的矩阵代数知识.

5.1.2　向量和矩阵

用 $\mathbf{R}^{m\times n}$ 表示全部 $m\times n$ 实矩阵的向量空间,$\mathbf{C}^{m\times n}$ 表示全部 $m\times n$ 复矩阵的向量空间.

$$\boldsymbol{A}\in\mathbf{R}^{m\times n}\Leftrightarrow\boldsymbol{A}=(a_{ij})=\begin{pmatrix}a_{11}&a_{12}&\cdots&a_{1n}\\a_{21}&a_{22}&\cdots&a_{2n}\\\vdots&\vdots&&\vdots\\a_{m1}&a_{m2}&\cdots&a_{mn}\end{pmatrix}$$

(实数排成的矩形表,称为 m 行 n 列矩阵).

$$\boldsymbol{x}\in\mathbf{R}^{n}\Leftrightarrow\boldsymbol{x}=\begin{pmatrix}x_1\\x_2\\\vdots\\x_n\end{pmatrix}\quad(n\text{ 维列向量}).$$

$$\boldsymbol{A}=(\boldsymbol{a}_1\quad\boldsymbol{a}_2\quad\cdots\quad\boldsymbol{a}_n),$$

其中 $\boldsymbol{a}_i$ 为 $\boldsymbol{A}$ 的第 i 列. 同理

$$\boldsymbol{A}=\begin{pmatrix}\boldsymbol{b}_1^{\mathrm{T}}\\\vdots\\\boldsymbol{b}_m^{\mathrm{T}}\end{pmatrix},$$

其中 $\boldsymbol{b}_i^{\mathrm{T}}$ 为 $\boldsymbol{A}$ 的第 i 行.

矩阵的基本运算:

(1) 矩阵加法　$\boldsymbol{C}=\boldsymbol{A}+\boldsymbol{B}$, $c_{ij}=a_{ij}+b_{ij}$ ($\boldsymbol{A}\in\mathbf{R}^{m\times n}$, $\boldsymbol{B}\in\mathbf{R}^{m\times n}$, $\boldsymbol{C}\in\mathbf{R}^{m\times n}$).

(2) 矩阵与标量的乘法　$\boldsymbol{C}=\alpha\boldsymbol{A}$, $c_{ij}=\alpha a_{ij}$.

(3) 矩阵与矩阵乘法 $\boldsymbol{C}=\boldsymbol{AB}$, $c_{ij}=\sum_{k=1}^{n}a_{ik}b_{kj}$ ($\boldsymbol{A}\in\mathbf{R}^{m\times n}$, $\boldsymbol{B}\in\mathbf{R}^{n\times p}$, $\boldsymbol{C}\in\mathbf{R}^{m\times p}$).

(4) 转置矩阵 $\boldsymbol{A}\in\mathbf{R}^{m\times n}$, $\boldsymbol{C}=\boldsymbol{A}^{\mathrm{T}}$, $c_{ij}=a_{ji}$.

(5) 单位矩阵 $\boldsymbol{I}=(\boldsymbol{e}_1\ \boldsymbol{e}_2\cdots\ \boldsymbol{e}_n)\in\mathbf{R}^{n\times n}$,其中

$$\boldsymbol{e}_k=(0,\cdots,0,1,0,\cdots,0)^{\mathrm{T}},\ k=1,2,\cdots,n.$$

(6) 非奇异矩阵 设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$, $\boldsymbol{B}\in\mathbf{R}^{n\times n}$. 如果 $\boldsymbol{AB}=\boldsymbol{BA}=\boldsymbol{I}$, 则称 $\boldsymbol{B}$ 是 $\boldsymbol{A}$ 的逆矩阵,记为 $\boldsymbol{A}^{-1}$, 且 $(\boldsymbol{A}^{-1})^{\mathrm{T}}=(\boldsymbol{A}^{\mathrm{T}})^{-1}$. 如果 $\boldsymbol{A}^{-1}$ 存在,则称 $\boldsymbol{A}$ 为非奇异矩阵. 如果 $\boldsymbol{A},\boldsymbol{B}\in\mathbf{R}^{n\times n}$均为非奇异矩阵,则 $(\boldsymbol{AB})^{-1}=\boldsymbol{B}^{-1}\boldsymbol{A}^{-1}$.

(7) 矩阵的行列式 设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$,则 $\boldsymbol{A}$ 的行列式可按任一行(或列)展开,即

$$\det(\boldsymbol{A})=\sum_{j=1}^{n}a_{ij}A_{ij}\qquad(i=1,2,\cdots,n),$$

其中 A_{ij} 为 a_{ij} 的代数余子式,$A_{ij}=(-1)^{i+j}M_{ij}$, M_{ij} 为元素 a_{ij} 的余子式.

行列式性质:

(a) $\det(\boldsymbol{AB})=\det(\boldsymbol{A})\det(\boldsymbol{B})$, $\boldsymbol{A}$, $\boldsymbol{B}\in\mathbf{R}^{n\times n}$.

(b) $\det(\boldsymbol{A}^{\mathrm{T}})=\det(\boldsymbol{A})$, $\boldsymbol{A}\in\mathbf{R}^{n\times n}$.

(c) $\det(c\boldsymbol{A})=c^n\det(\boldsymbol{A})$, $c\in\mathbf{R}$, $\boldsymbol{A}\in\mathbf{R}^{n\times n}$.

(d) $\det(\boldsymbol{A})\neq0\Leftrightarrow\boldsymbol{A}$ 是非奇异矩阵.

5.1.3 特殊矩阵

设 $\boldsymbol{A}=(a_{ij})\in\mathbf{R}^{n\times n}$.

(1) 对角矩阵 如果当 $i\neq j$ 时,$a_{ij}=0$.

(2) 三对角矩阵 如果当 $|i-j|>1$ 时,$a_{ij}=0$.

(3) 上三角矩阵 如果当 $i>j$ 时,$a_{ij}=0$.

(4) 上海森伯格(Hessenberg)阵 如果当 $i>j+1$ 时,a_{ij}

$=0$.

(5) 对称矩阵 如果 $\boldsymbol{A}^{\mathrm{T}}=\boldsymbol{A}$.

(6) 埃尔米特矩阵 设 $\boldsymbol{A}\in\mathbf{C}^{n\times n}$，如果 $\boldsymbol{A}^{\mathrm{H}}=\boldsymbol{A}$($\boldsymbol{A}^{\mathrm{H}}=\overline{\boldsymbol{A}}^{\mathrm{T}}$，即为 $\boldsymbol{A}$ 的共轭转置).

(7) 对称正定矩阵 如果(a) $\boldsymbol{A}^{\mathrm{T}}=\boldsymbol{A}$，(b) 对任意非零向量 $\boldsymbol{x}\in\mathbf{R}^{n}$，$(\boldsymbol{A}\boldsymbol{x},\boldsymbol{x})=\boldsymbol{x}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}>0$.

(8) 正交矩阵 如果 $\boldsymbol{A}^{-1}=\boldsymbol{A}^{\mathrm{T}}$.

(9) 酉矩阵 设 $\boldsymbol{A}\in\mathbf{C}^{n\times n}$，如果 $\boldsymbol{A}^{-1}=\boldsymbol{A}^{\mathrm{H}}$.

(10) 初等置换阵 由单位矩阵 $\boldsymbol{I}$ 交换第 i 行与第 j 行(或交换第 i 列与第 j 列)，得到的矩阵记为 $\boldsymbol{I}_{ij}$，且

$\boldsymbol{I}_{ij}\boldsymbol{A}=\widetilde{\boldsymbol{A}}$(为交换 $\boldsymbol{A}$ 第 i 行与第 j 行得到的矩阵)；

$\boldsymbol{A}\boldsymbol{I}_{ij}=\boldsymbol{B}$(为交换 $\boldsymbol{A}$ 第 i 列与第 j 列得到的矩阵).

(11) 置换阵 由初等置换阵的乘积得到的矩阵.

定理 1 设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$，则下述命题等价：

(1) 对任何 $\boldsymbol{b}\in\mathbf{R}^{n}$，方程组 $\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}$ 有唯一解.

(2) 齐次方程组 $\boldsymbol{A}\boldsymbol{x}=\boldsymbol{0}$ 只有唯一解 $\boldsymbol{x}=\boldsymbol{0}$.

(3) $\det(\boldsymbol{A})\neq 0$.

(4) $\boldsymbol{A}^{-1}$存在.

(5) $\boldsymbol{A}$ 的秩 $\mathrm{rank}(\boldsymbol{A})=n$.

定理 2 设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$为对称正定阵，则

(1) $\boldsymbol{A}$ 为非奇异矩阵，且 $\boldsymbol{A}^{-1}$亦是对称正定阵.

(2) 记 $\boldsymbol{A}_k$ 为 $\boldsymbol{A}$ 的顺序主子阵，则 $\boldsymbol{A}_k(k=1,2,\cdots,n)$亦是对称正定矩阵，其中

$$\boldsymbol{A}_k=\begin{pmatrix} a_{11} & \cdots & a_{1k} \\ \vdots & & \vdots \\ a_{k1} & \cdots & a_{kk} \end{pmatrix} \qquad (k=1,2,\cdots,n).$$

(3) $\boldsymbol{A}$ 的特征值 $\lambda_i>0(i=1,2,\cdots,n)$.

(4) $\boldsymbol{A}$ 的顺序主子式都大于零，即 $\det(\boldsymbol{A}_k)>0(k=1,2,\cdots,n)$.

定理 3 设 $A\in \mathbf{R}^{n\times n}$ 为对称矩阵. 如果 $\det(A_k)>0(k=1,2,\cdots,n)$,或 A 的特征值 $\lambda_i>0(i=1,2,\cdots,n)$,则 A 为对称正定阵.

有重特征值的矩阵不一定相似于对角矩阵,那么一般 n 阶矩阵 A 在相似变换下能简化到什么形状.

定理 4(Jordan 标准型) 设 A 为 n 阶矩阵,则存在一个非奇异矩阵 P 使得

$$P^{-1}AP=\begin{pmatrix} J_1(\lambda_1) & & & \\ & J_2(\lambda_2) & & \\ & & \ddots & \\ & & & J_r(\lambda_r) \end{pmatrix},$$

其中

$$J_i=\begin{pmatrix} \lambda_i & 1 & & & \\ & \ddots & \ddots & & \\ & & \lambda_i & \ddots & \\ & & & \ddots & 1 \\ & & & & \lambda_i \end{pmatrix}_{n_i\times n_i},$$

$$n_i\geqslant 1(i=1,2,\cdots,r),\ 且\sum_{i=1}^{r}n_i=n.$$

为**若当**(Jordan)**块**.

(1) 当 A 的若当标准型中所有若当块 J_i 均为一阶时,此标准型变成对角矩阵.

(2) 如果 A 的特征值各不相同,则其若当标准型必为对角阵 $\mathrm{diag}(\lambda_1,\lambda_2,\cdots,\lambda_n)$.

5.2 高斯消去法

本节介绍高斯消去法(逐次消去法)及消去法和矩阵三角分解

之间的关系. 虽然高斯消去法是一个古老的求解线性方程组的方法(早在公元前 250 年我国就掌握了解方程组的消去法),但由它改进、变形得到的选主元素消去法、三角分解法仍然是目前计算机上常用的有效方法.

5.2.1 高斯消去法

设有线性方程组

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1, \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2, \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m. \end{cases} \tag{2.1}$$

或写为矩阵形式

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix},$$

简记为 $\boldsymbol{Ax}=\boldsymbol{b}$.

首先举一个简单的例子来说明消去法的基本思想.

例 1　用消去法解方程组

$$\begin{cases} x_1 + x_2 + x_3 = 6, & (2.2) \\ 4x_2 - x_3 = 5, & (2.3) \\ 2x_1 - 2x_2 + x_3 = 1. & (2.4) \end{cases}$$

解　第 1 步. 将方程(2.2)乘上 −2 加到方程(2.4)上去,消去(2.4)中的未知数 x_1,得到

$$-4x_2 - x_3 = -11. \tag{2.5}$$

第 2 步. 将方程(2.3)加到方程(2.5)上去,消去方程(2.5)中的未知数 x_2,得到与原方程组等价的三角形方程组

$$\begin{cases} x_1 + x_2 + x_3 = 6, \\ 4x_2 - x_3 = 5, \\ -2x_3 = -6. \end{cases} \tag{2.6}$$

显然，方程组(2.6)是容易求解的，解为

$$x^* = (1,2,3)^T.$$

上述过程相当于

$$(\boldsymbol{A} \mid \boldsymbol{b}) = \begin{bmatrix} 1 & 1 & 1 & \vdots & 6 \\ 0 & 4 & -1 & \vdots & 5 \\ 2 & -2 & 1 & \vdots & 1 \end{bmatrix} \to \begin{bmatrix} 1 & 1 & 1 & \vdots & 6 \\ 0 & 4 & -1 & \vdots & 5 \\ 0 & -4 & -1 & \vdots & -11 \end{bmatrix}$$

$$\to \begin{bmatrix} 1 & 1 & 1 & \vdots & 6 \\ 0 & 4 & -1 & \vdots & 5 \\ 0 & 0 & -2 & \vdots & -6 \end{bmatrix}$$

$$(-2) \times r_1 + r_3 \to r_3 \quad r_2 + r_3 \to r_3$$

其中用 r_i 表示矩阵的第 i 行.

由此看出，用消去法解方程组的基本思想是用逐次消去未知数的方法把原方程组 $\boldsymbol{Ax}=\boldsymbol{b}$ 化为与其等价的三角形方程组，而求解三角形方程组可用回代的方法求解. 换句话说，上述过程就是用行的初等变换将原方程组系数矩阵化为简单形式(上三角矩阵)，从而将求解原方程组(2.1)的问题转化为求解简单方程组的问题. 或者说，对系数矩阵 $\boldsymbol{A}$ 施行一些左变换(用一些简单矩阵)将其约化为上三角矩阵.

下面我们讨论求解一般线性方程组的高斯消去法.

将(2.1)记为 $\boldsymbol{A}^{(1)}\boldsymbol{x}=\boldsymbol{b}^{(1)}$，其中

$$\boldsymbol{A}^{(1)} = (a_{ij}^{(1)}) = (a_{ij}), \ \boldsymbol{b}^{(1)} = \boldsymbol{b}.$$

(1) 第 1 步($k=1$).

设 $a_{11}^{(1)} \neq 0$，首先计算乘数

$$m_{i1} = a_{i1}^{(1)}/a_{11}^{(1)} \quad (i = 2,3,\cdots,m).$$

用 $-m_{i1}$ 乘(2.1)的第一个方程，加到第 i 个($i=2,3,\cdots,m$)方程

上，消去(2.1)的从第二个方程到第 m 个方程中的未知数 x_1，得到与(2.1)等价的方程组

$$\begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{m2}^{(2)} & \cdots & a_{mn}^{(2)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_m^{(2)} \end{pmatrix}. \tag{2.7}$$

简记为

$$A^{(2)} x = b^{(2)},$$

其中 $A^{(2)}$，$b^{(2)}$ 的元素计算公式为

$$\begin{cases} a_{ij}^{(2)} = a_{ij}^{(1)} - m_{i1} a_{1j}^{(1)} & (i = 2, \cdots, m;\ j = 2, \cdots, n), \\ b_i^{(2)} = b_i^{(1)} - m_{i1} b_1^{(1)} & (i = 2, \cdots, m). \end{cases}$$

(2) 第 k 次消元($k=1,2,\cdots,s=\min(m-1,n)$)．

设上述第 1 步，…，第 $k-1$ 步消元过程计算已经完成，即已计算好与(2.1)等价的方程组

$$\begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1k}^{(1)} & \cdots & a_{1n}^{(1)} \\ & a_{22}^{(2)} & \cdots & a_{2k}^{(2)} & \cdots & a_{2n}^{(2)} \\ & & \ddots & \vdots & & \vdots \\ & & & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} \\ & & & \vdots & & \vdots \\ & & & a_{mk}^{(k)} & \cdots & a_{mn}^{(k)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_k^{(k)} \\ \vdots \\ b_m^{(k)} \end{pmatrix}, \tag{2.8}$$

简记为 $A^{(k)} x = b^{(k)}$．

设 $a_{kk}^{(k)} \neq 0$，计算乘数

$$m_{ik} = a_{ik}^{(k)} / a_{kk}^{(k)} \quad (i = k+1, \cdots, m).$$

用 $-m_{ik}$ 乘(2.8)的第 k 个方程加到第 i 个方程($i=k+1,\cdots,m$)，消去从第 $k+1$ 个方程到第 m 个方程中的未知数 x_k，得到与(2.1)等价的方程组 $A^{(k+1)} x = b^{(k+1)}$．$A^{(k+1)}$，$b^{(k+1)}$ 元素的计算公式为

$$\begin{cases} a_{ij}^{(k+1)} = a_{ij}^{(k)} - m_{ik} a_{kj}^{(k)} & (i = k+1, \cdots, m;\ j = k+1, \cdots, n), \\ b_i^{(k+1)} = b_i^{(k)} - m_{ik} b_k^{(k)} & (i = k+1, \cdots, m). \end{cases} \tag{2.9}$$

显然 $\boldsymbol{A}^{(k+1)}$ 中从第 1 行到第 k 行与 $\boldsymbol{A}^{(k)}$ 相同.

(3) 继续上述过程,且设 $a_{kk}^{(k)}\neq 0(i=1,2,\cdots,s)$,直到完成第 s 步消元计算. 最后得到与原方程组等价的简单方程组 $\boldsymbol{A}^{(s+1)}\boldsymbol{x}=\boldsymbol{b}^{(s+1)}$,其中 $\boldsymbol{A}^{(s+1)}$ 为上梯形.

特别当 $m=n$ 时,与原方程组等价的方程组为 $\boldsymbol{A}^{(n)}\boldsymbol{x}=\boldsymbol{b}^{(n)}$,即

$$\begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} \\ & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ & & \ddots & \vdots \\ & & & a_{nn}^{(n)} \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}=\begin{pmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_n^{(n)} \end{pmatrix}. \tag{2.10}$$

由(2.1)约化为(2.10)的过程称为消元过程.

如果 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 是非奇异矩阵,且 $a_{kk}^{(k)}\neq 0(k=1,2,\cdots,n-1)$,求解三角形方程组(2.10),得到求解公式

$$\begin{cases} x_n = b^{(n)}/a_{nn}^{(n)}, \\ x_k = \left(b_k^{(k)} - \sum\limits_{j=k+1}^{n} a_{kj}^{(k)} x_j\right)/a_{kk}^{(k)} \quad (k=n-1,n-2,\cdots,1). \end{cases} \tag{2.11}$$

(2.10)的求解过程(2.11)称为回代过程.

注意:设 $\boldsymbol{Ax}=\boldsymbol{b}$,其中 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 为非奇异矩阵,如果 $a_{11}=0$,由于 $\boldsymbol{A}$ 为非奇异矩阵,所以 $\boldsymbol{A}$ 的第一列一定有元素不等于零,例如 $a_{i_1 1}\neq 0$,于是可交换两行元素(即 $r_1\longleftrightarrow r_{i_1}$),将 $a_{i_1 1}$ 调到(1,1)位置,然后进行消元计算,这时 $\boldsymbol{A}^{(2)}$ 右下角矩阵为 $n-1$ 阶非奇异矩阵. 继续这过程,高斯消去法照样可进行计算.

总结上述讨论即有

定理 5 设 $\boldsymbol{Ax}=\boldsymbol{b}$,其中 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$.

(1) 如果 $a_{kk}^{(k)}\neq 0(k=1,2,\cdots,n)$,则可通过高斯消去法将 $\boldsymbol{Ax}=\boldsymbol{b}$ 约化为等价的三角形方程组(2.10),且计算公式为:

(a) 消元计算$(k=1,2,\cdots,n-1)$

$$\begin{cases} m_{ik} = a_{ik}^{(k)}/a_{kk}^{(k)} & (i = k+1,\cdots,n), \\ a_{ij}^{(k+1)} = a_{ij}^{(k)} - m_{ik}a_{kj}^{(k)} & (i,j = k+1,\cdots,n), \\ b_i^{(k+1)} = b_i^{(k)} - m_{ik}b_k^{(k)} & (i = k+1,\cdots,n). \end{cases}$$

(b) 回代计算

$$\begin{cases} x_n = b_n^{(n)}/a_{nn}^{(n)}, \\ x_i = \left(b_i^{(i)} - \sum_{j=i+1}^{n} a_{ij}^{(i)}x_j\right)/a_{ii}^{(i)} & (i = n-1,\cdots,2,1). \end{cases}$$

(2) 如果 $\boldsymbol{A}$ 为非奇异矩阵，则可通过高斯消去法(及交换两行的初等变换)将方程组 $\boldsymbol{Ax}=\boldsymbol{b}$ 约化为(2.10).

算法 1(高斯算法) 设 $\mathbf{A}\in\mathbf{R}^{m\times n}(m>1)$, $s=\min(m-1,n)$, 如果 $a_{kk}^{(k)}\neq 0(k=1,2,\cdots,s)$，本算法用高斯方法将 $\boldsymbol{A}$ 约化为上梯形，且 $\mathbf{A}^{(k)}$ 覆盖 $\boldsymbol{A}$，乘数 m_{ik} 覆盖 a_{ik}.

对于 $k=1,2,\cdots,s$

(1) 如果 $a_{kk}=0$，则计算停止

(2) 对于 $i=k+1,\cdots,m$

(a) $a_{ik}\leftarrow m_{ik}=a_{ik}/a_{kk}$

(b) 对于 $j=k+1,\cdots,n$

$$a_{ij}\leftarrow a_{ij}-m_{ik} * a_{kj}.$$

显然，算法 1 第 k 步需要作 $m-k$ 次除法，$(m-k)(n-k)$次乘法运算，因此，本算法(从第 1 步到第 s 步消元计算总的计算量)大约需要 $s^3/3-(m+n)s^2/2+mns$ 次乘法运算(对相当大的 s). 当 $m=n$ 时，总共大约需要 $n^3/3$ 次乘法运算.

数 $a_{kk}^{(k)}$ 在高斯消去法中有着突出的作用，称为约化的**主元素**.

算法 2(回代算法) 设 $\boldsymbol{Ux}=\boldsymbol{b}$，其中 $\boldsymbol{U}\in\mathbf{R}^{n\times n}$ 为非奇异上三角阵，本算法计算 $\boldsymbol{Ux}=\boldsymbol{b}$ 的解.

对于 $i=n,\cdots,1$

(1) $x_i\leftarrow b_i$

(2) 对于 $j=i+1,\cdots,n$

$$x_i \leftarrow x_i - u_{ij} * x_j$$

(3) $x_i \leftarrow x_i / u_{ii}$

这个算法需要 $n(n+1)/2$ 乘除法运算.

高斯消去法对于某些简单的矩阵可能会失败,例如

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

由此,需要对算法 1 进行修改,首先研究原来矩阵 $\mathbf{A}$ 在什么条件下才能保证 $a_{kk}^{(k)} \neq 0(k=1,2,\cdots)$. 下面的定理给出了这个条件.

定理 6 约化的主元素 $a_{ii}^{(i)} \neq 0(i=1,2,\cdots,k)$的充要条件是矩阵 $\mathbf{A}$ 的顺序主子式 $D_i \neq 0(i=1,2,\cdots,k)$. 即

$$D_1 = a_{11} \neq 0,$$

$$D_i = \begin{vmatrix} a_{11} & \cdots & a_{1i} \\ \vdots & & \vdots \\ a_{i1} & \cdots & a_{ii} \end{vmatrix} \neq 0 \quad (i = 1,2,\cdots,k). \tag{2.12}$$

证明 首先利用归纳法证明定理 6 的充分性. 显然,当 $k=1$ 时,定理 6 成立,现设定理 6 充分性对 $k-1$ 是成立的,求证定理 6 充分性对 k 亦成立. 设 $D_i \neq 0(i=1,2,\cdots,k)$,于是由归纳法假设有 $a_{ii}^{(i)} \neq 0(i=1,2,\cdots,k-1)$,可用高斯消去法将 $\mathbf{A}^{(1)}$ 约化到 $\mathbf{A}^{(k)}$,即

$$\mathbf{A}^{(1)} \to \mathbf{A}^{(k)} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1k}^{(1)} & \cdots & a_{1n}^{(1)} \\ & a_{22}^{(2)} & \cdots & a_{2k}^{(2)} & \cdots & a_{2n}^{(2)} \\ & & \ddots & \vdots & & \vdots \\ & & & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} \\ & & & \vdots & & \vdots \\ & & & a_{nk}^{(k)} & \cdots & a_{nn}^{(k)} \end{pmatrix},$$

且有

$$D_2 = \begin{vmatrix} a_{11}^{(1)} & a_{12}^{(1)} \\ 0 & a_{22}^{(2)} \end{vmatrix} = a_{11}^{(1)} a_{22}^{(2)},$$

$$\cdots \qquad \cdots$$

$$D_k = \begin{vmatrix} a_{11}^{(1)} & \cdots & a_{1k}^{(1)} \\ & \ddots & \vdots \\ & & a_{kk}^{(k)} \end{vmatrix} = a_{11}^{(1)} a_{22}^{(2)} \cdots a_{kk}^{(k)}. \tag{2.13}$$

由设 $D_i \neq 0(i=1,2,\cdots,k)$，利用(2.13)式，则有 $a_{kk}^{(k)} \neq 0$，定理 6 充分性对 k 亦成立.

显然，由假设 $a_{ii}^{(i)} \neq 0(i=1,2,\cdots,k)$，利用(2.13)式亦可推出 $D_i \neq 0(i=1,2,\cdots,k)$.

推论　如果 $\boldsymbol{A}$ 的顺序主子式 $D_k \neq 0(k=1,2,\cdots,n-1)$，则

$$\begin{cases} a_{11}^{(1)} = D_1, \\ a_{kk}^{(k)} = D_k / D_{k-1} \qquad (k = 2,3,\cdots,n). \end{cases}$$

5.2.2　矩阵的三角分解

下面我们借助矩阵理论进一步对消去法作些分析，从而建立高斯消去法与矩阵因式分解的关系.

设(2.1)的系数矩阵 $\boldsymbol{A} \in \mathbf{R}^{n \times n}$ 的各顺序主子式均不为零. 由于对 $\boldsymbol{A}$ 施行行的初等变换相当于用初等矩阵左乘 $\boldsymbol{A}$，于是对(2.1)施行第一次消元后化为(2.7)，这时 $\boldsymbol{A}^{(1)}$ 化为 $\boldsymbol{A}^{(2)}$，$\boldsymbol{b}^{(1)}$ 化为 $\boldsymbol{b}^{(2)}$，即

$$\boldsymbol{L}_1 \boldsymbol{A}^{(1)} = \boldsymbol{A}^{(2)}, \ \boldsymbol{L}_1 \boldsymbol{b}^{(1)} = \boldsymbol{b}^{(2)},$$

其中

$$\boldsymbol{L}_1 = \begin{pmatrix} 1 & & & & \\ -m_{21} & 1 & & & \\ -m_{31} & & 1 & & \\ \vdots & & & \ddots & \\ -m_{n1} & & & & 1 \end{pmatrix}.$$

一般第 k 步消元，$\boldsymbol{A}^{(k)}$ 化为 $\boldsymbol{A}^{(k+1)}$，$\boldsymbol{b}^{(k)}$ 化为 $\boldsymbol{b}^{(k+1)}$，相当于

$$\boldsymbol{L}_k\boldsymbol{A}^{(k)} = \boldsymbol{A}^{(k+1)},\ \boldsymbol{L}_k\boldsymbol{b}^{(k)} = \boldsymbol{b}^{(k+1)},$$

其中

$$\boldsymbol{L}_k = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & -m_{k+1,k} & 1 & & \\ & & \vdots & & \ddots & \\ & & -m_{nk} & & & 1 \end{pmatrix}.$$

重复这过程，最后得到

$$\left.\begin{aligned} \boldsymbol{L}_{n-1}\cdots\boldsymbol{L}_2\boldsymbol{L}_1\boldsymbol{A}^{(1)} &= \boldsymbol{A}^{(n)}; \\ \boldsymbol{L}_{n-1}\cdots\boldsymbol{L}_2\boldsymbol{L}_1\boldsymbol{b}^{(1)} &= \boldsymbol{b}^{(n)}. \end{aligned}\right\} \tag{2.14}$$

将上三角矩阵 $\boldsymbol{A}^{(n)}$ 记为 $\boldsymbol{U}$，由(2.14)得到

$$\boldsymbol{A} = \boldsymbol{L}_1^{-1}\boldsymbol{L}_2^{-1}\cdots\boldsymbol{L}_{n-1}^{-1}\boldsymbol{U} = \boldsymbol{L}\boldsymbol{U},$$

其中

$$\boldsymbol{L} = \boldsymbol{L}_1^{-1}\boldsymbol{L}_2^{-1}\cdots\boldsymbol{L}_{n-1}^{-1} = \begin{pmatrix} 1 & & & & \\ m_{21} & 1 & & & \\ m_{31} & m_{32} & 1 & & \\ \vdots & \vdots & \vdots & \ddots & \\ m_{n1} & m_{n2} & m_{n3} & \cdots & 1 \end{pmatrix}$$

为单位下三角矩阵.

这就是说，高斯消去法实质上产生了一个将 $\boldsymbol{A}$ 分解为两个三角形矩阵相乘的因式分解，于是我们得到如下重要定理，它在解方程组的直接法中起着重要作用.

定理 7（矩阵的 LU 分解） 设 $\boldsymbol{A}$ 为 n 阶矩阵，如果 $\boldsymbol{A}$ 的顺序主子式 $D_i \neq 0 (i=1,2,\cdots,n-1)$，则 $\boldsymbol{A}$ 可分解为一个单位下三角矩阵 $\boldsymbol{L}$ 和一个上三角矩阵 $\boldsymbol{U}$ 的乘积，且这种分解是唯一的.

证明 根据以上高斯消去法的矩阵分析，$\boldsymbol{A}=\boldsymbol{L}\boldsymbol{U}$ 的存在性已经得到证明，现仅在 $\boldsymbol{A}$ 为非奇异矩阵的假定下来证明唯一性，当 $\boldsymbol{A}$ 为奇异矩阵的情况留作练习. 设

$$\boldsymbol{A}=\boldsymbol{L}\boldsymbol{U}=\boldsymbol{L}_1\boldsymbol{U}_1,$$

其中 $\boldsymbol{L},\boldsymbol{L}_1$ 为单位下三角矩阵，$\boldsymbol{U},\boldsymbol{U}_1$ 为上三角矩阵.

由于 $\boldsymbol{U}_1^{-1}$ 存在，故

$$\boldsymbol{L}^{-1}\boldsymbol{L}_1=\boldsymbol{U}\boldsymbol{U}_1^{-1}.$$

上式右边为上三角矩阵，左边为单位下三角矩阵，从而上式两边都必须等于单位矩阵，故 $\boldsymbol{U}=\boldsymbol{U}_1$，$\boldsymbol{L}=\boldsymbol{L}_1$. 证毕.

例 2 对于例 1，系数矩阵

$$\boldsymbol{A}=\begin{pmatrix}1 & 1 & 1\\ 0 & 4 & -1\\ 2 & -2 & 1\end{pmatrix},$$

由高斯消去法，$m_{21}=0$，$m_{31}=2$，$m_{32}=-1$，故

$$\boldsymbol{A}=\begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 2 & -1 & 1\end{pmatrix}\begin{pmatrix}1 & 1 & 1\\ 0 & 4 & -1\\ 0 & 0 & -2\end{pmatrix}=\boldsymbol{L}\boldsymbol{U}.$$

5.3 高斯主元素消去法

由高斯消去法知道，在消元过程中可能出现 $a_{kk}^{(k)}=0$ 的情况，这时消去法将无法进行；即使主元素 $a_{kk}^{(k)}\neq 0$ 但很小时，用其作除数，会导致其他元素数量级的严重增长和舍入误差的扩散，最后也使得计算解不可靠.

例 3 求解方程组.

$$\begin{pmatrix}0.001 & 2.000 & 3.000\\ -1.000 & 3.712 & 4.623\\ -2.000 & 1.072 & 5.643\end{pmatrix}\begin{pmatrix}x_1\\ x_2\\ x_3\end{pmatrix}=\begin{pmatrix}1.000\\ 2.000\\ 3.000\end{pmatrix}.$$

用 4 位浮点数进行计算. 精确解舍入到 4 位有效数字为

$$x^* = (-0.4904, -0.05104, 0.3675)^T,$$

解 (方法 1)用高斯消去法求解.

$$(\boldsymbol{A} \mid \boldsymbol{b}) = \begin{bmatrix} 0.001 & 2.000 & 3.000 & \vdots & 1.000 \\ -1.000 & 3.712 & 4.623 & \vdots & 2.000 \\ -2.000 & 1.072 & 5.643 & \vdots & 3.000 \end{bmatrix} \begin{matrix} m_{21} = -1.000/0.001 \\ = -1000 \\ m_{31} = -2.000/0.001 \\ = -2000 \end{matrix}$$

$$\to \begin{bmatrix} 0.001 & 2.000 & 3.000 & \vdots & 1.000 \\ 0 & 2004 & 3005 & \vdots & 1002 \\ 0 & 4001 & 6006 & \vdots & 2003 \end{bmatrix} \begin{matrix} m_{32} = 4001/2004 \\ = 1.997 \end{matrix}$$

$$\to \begin{bmatrix} 0.001 & 2.000 & 3.000 & \vdots & 1.000 \\ 0 & 2004 & 3005 & \vdots & 1002 \\ 0 & 0 & 5.000 & \vdots & 2.000 \end{bmatrix},$$

计算解为

$$\bar{x} = (-0.400, -0.09980, 0.4000)^T.$$

显然计算解 $\bar{x}$ 是一个很坏的结果,不能作为方程组的近似解. 其原因是我们在消元计算时用了小主元 0.001,使得约化后的方程组元素数量级大大增长,经再舍入使得在计算(3,3)元素时发生了严重的相消情况((3,3)元素舍入到第 4 位数字的正确值是 5.922),因此经消元后得到的三角形方程组就不准确了.

(方法 2)交换行,避免绝对值小的主元作除数.

$$(\boldsymbol{A} \mid \boldsymbol{b}) \xrightarrow[r_1 \leftrightarrow r_3]{} \begin{bmatrix} -2.000 & 1.072 & 5.643 & \vdots & 3.000 \\ -1.000 & 3.712 & 4.623 & \vdots & 2.000 \\ 0.001 & 2.000 & 3.000 & \vdots & 1.000 \end{bmatrix} \begin{matrix} m_{21} = 0.5000 \\ m_{31} = -0.0005 \end{matrix}$$

$$\to \begin{bmatrix} -2.000 & 1.072 & 5.643 & \vdots & 3.000 \\ 0 & 3.176 & 1.801 & \vdots & 0.5000 \\ 0 & 2.001 & 3.003 & \vdots & 1.002 \end{bmatrix} m_{32} = 0.6300$$

$$\rightarrow \left(\begin{array}{ccc|c} -2.000 & 1.072 & 5.643 & 3.000 \\ 0 & 3.176 & 1.801 & 0.5000 \\ 0 & 0 & 1.868 & 0.6870 \end{array}\right),$$

得计算解为

$$x = (-0.4900, -0.05113, 0.3678)^{T} \approx x^{*}.$$

这个例子告诉我们,在采用高斯消去法解方程组时,小主元可能产生麻烦,故应避免采用绝对值小的主元素 $a_{kk}^{(k)}$. 对一般矩阵来说,最好每一步选取系数矩阵(或消元后的低阶矩阵)中绝对值最大的元素作为主元素,以使高斯消去法具有较好的数值稳定性.这就是全主元素消去法,在选主元时要花费较多机器时间,目前主要使用的是列主元消去法,本节主要介绍列主元消去法,并假定(2.1)的 $A \in \mathbb{R}^{n\times n}$ 为非奇异的.

5.3.1　列主元素消去法

设方程组(2.1)的增广矩阵为

$$B = \left(\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{array}\right).$$

首先在 A 的第一列中选取绝对值最大的元素作为主元素,例如

$$|a_{i_1,1}| = \max_{1\leqslant i\leqslant n} |a_{i1}| \neq 0,$$

然后交换 B 的第1行与第 i_1 行,经第1次消元计算得

$$(A \mid b) \rightarrow (A^{(2)} \mid b^{(2)}).$$

重复上述过程,设已完成第 $k-1$ 步的选主元素,交换两行及消元计算,$(A|b)$约化为

$$(\boldsymbol{A}^{(k)} \mid \boldsymbol{b}^{(k)}) = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1k} & \cdots & a_{1n} & \vdots & b_1 \\ & a_{22} & \cdots & a_{2k} & \cdots & a_{2n} & \vdots & b_2 \\ & & \ddots & \vdots & & \vdots & \vdots & \vdots \\ & & & a_{kk} & \cdots & a_{kn} & \vdots & b_k \\ & & & \vdots & & \vdots & \vdots & \vdots \\ & & & a_{nk} & \cdots & a_{nn} & \vdots & b_n \end{pmatrix},$$

其中 $\boldsymbol{A}^{(k)}$ 的元素仍记为 a_{ij}，$\boldsymbol{b}^{(k)}$ 的元素仍记为 b_i.

第 k 步选主元素（在 $\boldsymbol{A}^{(k)}$ 右下角方阵的第 1 列内选），即确定 i_k，使

$$|a_{i_k,k}| = \max_{k \leqslant i \leqslant n} |a_{ik}| \neq 0.$$

交换 $(\boldsymbol{A}^{(k)} \mid \boldsymbol{b}^{(k)})$ 第 k 行与 i_k 行的元素，再进行消元计算，最后将原方程组化为 $(k=1,2,\cdots,n-1)$

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ & a_{22} & \cdots & a_{2n} \\ & & \ddots & \vdots \\ & & & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}.$$

回代求解

$$\begin{cases} x_n = b_n / a_{nn}; \\ x_i = \left(b_i - \sum\limits_{j=i+1}^{n} a_{ij} x_j\right) / a_{ii} \quad (i = n-1, \cdots, 2, 1). \end{cases}$$

算法 3（列主元素消去法） 设 $\boldsymbol{Ax}=\boldsymbol{b}$. 本算法用 $\boldsymbol{A}$ 的具有行交换的列主元素消去法，消元结果冲掉 $\boldsymbol{A}$，乘数 m_{ij} 冲掉 a_{ij}，计算解 $\boldsymbol{x}$ 冲掉常数项 $\boldsymbol{b}$，行列式存放在 det 中.

1. det←1
2. 对于 $k=1,2,\cdots,n-1$

   (1) 按列选主元

$$|a_{i_k,k}| = \max_{k \leq i \leq n} |a_{ik}|$$

(2) 如果 $a_{i_k,k}=0$,则计算停止($\det(\boldsymbol{A})=0$)

(3) 如果 $i_k=k$ 则转(4)

换行:$a_{kj} \longleftrightarrow a_{i_k,j} (j=k,k+1,\cdots,n)$

$b_k \longleftrightarrow b_{i_k}$

$\det \leftarrow -\det$

(4) 消元计算

对于 $i=k+1,\cdots,n$

(a) $a_{ik} \leftarrow m_{ik} = a_{ik}/a_{kk}$

(b) 对于 $j=k+1,\cdots,n$

$$a_{ij} \leftarrow a_{ij} - m_{ik} * a_{kj}$$

(c) $b_i \leftarrow b_i - m_{ik} * b_k$

(5) $\det \leftarrow a_{kk} * \det$

3. 如果 $a_{nn}=0$,则计算停止($\det(\boldsymbol{A})=0$)

4. 回代求解

(1) $b_n \leftarrow b_n/a_{nn}$

(2) 对于 $i=n-1,\cdots,2,1$

$$b_i \leftarrow \left(b_i - \sum_{j=i+1}^{n} a_{ij} * b_j\right)/a_{ii}$$

5. $\det \leftarrow a_{nn} * \det$

例 3 的(方法 2)用的就是列主元素消去法.

下面用矩阵运算来描述解(2.1)的列主元素消去法. 列主元素消去法为

$$\left.\begin{array}{l} \boldsymbol{L}_1 \boldsymbol{I}_{1,i_1} \boldsymbol{A}^{(1)} = \boldsymbol{A}^{(2)},\ \boldsymbol{L}_1 \boldsymbol{I}_{1,i_1} \boldsymbol{b}^{(1)} = \boldsymbol{b}^{(2)}, \\ \boldsymbol{L}_k \boldsymbol{I}_{k,i_k} \boldsymbol{A}^{(k)} = \boldsymbol{A}^{(k+1)},\ \boldsymbol{L}_k \boldsymbol{I}_{k,i_k} \boldsymbol{b}^{(k)} = \boldsymbol{b}^{(k+1)}. \end{array}\right\} \tag{3.1}$$

其中 $\boldsymbol{L}_k$ 的元素满足 $|m_{ik}| \leqslant 1 (k=1,2,\cdots,n-1)$, $\boldsymbol{I}_{k,i_k}$ 是初等置换阵.

利用(3.1)得到

$$L_{n-1}I_{n-1,i_{n-1}}\cdots L_2I_{2,i_2}L_1I_{1,i_1}A=A^{(n)}=U.$$

简记为

$$\widetilde{P}A=U,\ \widetilde{P}b=b^{(n)},$$

其中

$$\widetilde{P}=L_{n-1}I_{n-1,i_{n-1}}\cdots L_2I_{2,i_2}L_1I_{1,i_1}.$$

下面就 $n=4$ 来考察一下矩阵 $\widetilde{P}$.

$$\begin{aligned}U=A^{(4)}&=L_3I_{3,i_3}L_2I_{2,i_2}L_1I_{1,i_1}A\\&=L_3(I_{3,i_3}L_2I_{3,i_3})(I_{3,i_3}I_{2,i_2}L_1I_{2,i_2}I_{3,i_3})\\&\quad\cdot(I_{3,i_3}I_{2,i_2}I_{1,i_1})A\equiv\widetilde{L}_3\widetilde{L}_2\widetilde{L}_1PA,\end{aligned}\tag{3.2}$$

其中

$$\begin{aligned}\widetilde{L}_1&=I_{3,i_3}I_{2,i_2}L_1I_{2,i_2}I_{3,i_3},\\\widetilde{L}_2&=I_{3,i_3}L_2I_{3,i_3},\\\widetilde{L}_3&=L_3,\\P&=I_{3,i_3}I_{2,i_2}I_{1,i_1}.\end{aligned}$$

由习题 3 知 $\widetilde{L}_k(k=1,2,3)$亦为单位下三角阵,其元素的绝对值不超过 1. 记

$$L^{-1}=\widetilde{L}_3\widetilde{L}_2\widetilde{L}_1,$$

由(3.2)得到

$$PA=LU,$$

其中 P 为排列矩阵,L 为单位下三角阵,U 为上三角阵. 这说明对(2.1)应用列主元素消去法相当于对$(A|b)$先进行一系列行交换后对 $PAx=Pb$ 再应用高斯消去法. 在实际计算中我们只能在计算过程中做行的交换.

总结以上的讨论有

定理 8(列主元素的三角分解定理) 如果 A 为非奇异矩阵,

则存在排列矩阵 $\boldsymbol{P}$ 使

$$\boldsymbol{PA} = \boldsymbol{LU},$$

其中 $\boldsymbol{L}$ 为单位下三角阵，$\boldsymbol{U}$ 为上三角阵.

在编程实现过程中，$\boldsymbol{L}$ 元素存放在数组 $\boldsymbol{A}$ 的下三角部分，$\boldsymbol{U}$ 元素存放在 $\boldsymbol{A}$ 上三角部分，由记录主行的整型数组 Ip(n)可知 $\boldsymbol{P}$ 的情况.

5.3.2　高斯-若当消去法

高斯消去法始终是消去对角线下方的元素，现考虑高斯消去法的一种修正，即消去对角线下方和上方的元素，这种方法称为高斯-若当(Gauss-Jordan)消去法.

设用高斯-若当消去法已完成 $k-1$ 步，于是 $\boldsymbol{Ax}=\boldsymbol{b}$ 化为等价方程组 $\boldsymbol{A}^{(k)}\boldsymbol{x}=\boldsymbol{b}^{(k)}$，其中

$$(\boldsymbol{A}^{(k)} \mid \boldsymbol{b}^{(k)}) = \left(\begin{array}{cccccccc:c} 1 & 0 & \cdots & 0 & a_{1k} & \cdots & a_{1n} & & b_1 \\ & 1 & \cdots & 0 & a_{2k} & \cdots & a_{2n} & & b_2 \\ & & \ddots & \vdots & \vdots & & \vdots & & \vdots \\ & & & 1 & a_{k-1,k} & \cdots & a_{k-1,n} & & b_{k-1} \\ & & & & a_{kk} & \cdots & a_{kn} & & b_k \\ & & & & \vdots & & \vdots & & \vdots \\ & & & & a_{nk} & \cdots & a_{nn} & & b_n \end{array}\right).$$

在第 k 步计算时($k=1,2,\cdots,n$)，考虑对上述矩阵的第 k 行上、下都进行消元计算.

1. 按列选主元素，即确定 i_k 使

$$|a_{i_k,k}| = \max_{k\leqslant i\leqslant n} |a_{ik}|.$$

2. 换行(当 $i_k \neq k$ 时)交换$(\boldsymbol{A}|\boldsymbol{b})$第 k 行与第 i_k 行元素.

3. 计算乘数　$m_{ik} = -a_{ik}/a_{kk}$　($i=1,2,\cdots,n$ 且 $i\neq k$)，

$$m_{kk} = 1/a_{kk}.$$

(m_{ik} 可保存在存放 a_{ik} 的单元中).

4. 消元计算

$$a_{ij} \leftarrow a_{ij} + m_{ik}a_{kj} \quad \begin{pmatrix} i = 1,2,\cdots,n \text{ 且 } i \neq k \\ j = k+1,\cdots,n \end{pmatrix},$$

$$b_i \leftarrow b_i + m_{ik}b_k \quad (i = 1,2,\cdots,n \text{ 且 } i \neq k).$$

5. 计算主行

$$a_{kj} \leftarrow a_{kj}m_{kk} \quad (j = k,k+1,\cdots,n),$$

$$b_k \leftarrow b_k m_{kk}.$$

上述过程结束后有

$$(\boldsymbol{A} \mid \boldsymbol{b}) \rightarrow (\boldsymbol{A}^{(k+1)} \mid \boldsymbol{b}^{(k+1)}) = \left(\begin{array}{cccc:c} 1 & & & & \hat{b}_1 \\ & 1 & & & \hat{b}_2 \\ & & \ddots & & \vdots \\ & & & 1 & \hat{b}_n \end{array}\right).$$

说明用高斯-若当方法将 $\boldsymbol{A}$ 约化为单位矩阵，计算解就在常数项位置得到，因此用不着回代求解，用高斯-若当方法解方程组其计算量大约需要 $n^3/2$ 次乘除法，要比高斯消去法大，但用高斯-若当方法求一个矩阵的逆矩阵还是比较合适的.

定理 9(高斯-若当法求逆矩阵) 设 $\boldsymbol{A}$ 为非奇异矩阵，方程组 $\boldsymbol{AX}=\boldsymbol{I}_n$ 的增广矩阵为 $\boldsymbol{C}=(\boldsymbol{A}\mid\boldsymbol{I}_n)$. 如果对 $\boldsymbol{C}$ 应用高斯-若当方法化为 $(\boldsymbol{I}_n\mid\boldsymbol{T})$，则 $\boldsymbol{A}^{-1}=\boldsymbol{T}$.

事实上，求 $\boldsymbol{A}$ 的逆矩阵 $\boldsymbol{A}^{-1}$，即求 n 阶矩阵 $\boldsymbol{X}$，使 $\boldsymbol{AX}=\boldsymbol{I}_n$，其中 $\boldsymbol{I}_n$ 为单位矩阵. 将 $\boldsymbol{X}$ 按列分块

$$\boldsymbol{X} = (\boldsymbol{x}_1\ \boldsymbol{x}_2\ \cdots\ \boldsymbol{x}_n),\ \boldsymbol{I} = (\boldsymbol{e}_1\ \boldsymbol{e}_2\ \cdots\ \boldsymbol{e}_n),$$

于是求解 $\boldsymbol{AX}=\boldsymbol{I}_n$ 等价于求解 n 个方程组

$$\boldsymbol{Ax}_j = \boldsymbol{e}_j \quad (j = 1,2,\cdots,n).$$

我们可用高斯-若当方法求解 $\boldsymbol{AX}=\boldsymbol{I}_n$.

例 4 用高斯-若当方法求

$$A=\begin{pmatrix}1&2&3\\2&4&5\\3&5&6\end{pmatrix}$$

的逆矩阵 A^{-1}.

解 $C=\left(\begin{array}{ccc:ccc}1&2&3&1&0&0\\2&4&5&0&1&0\\3&5&6&0&0&1\end{array}\right)\xrightarrow{r_1\leftrightarrow r_3}\left(\begin{array}{ccc:ccc}3&5&6&0&0&1\\2&4&5&0&1&0\\1&2&3&1&0&0\end{array}\right)$

$$\xrightarrow{\text{第 1 次消元}}\left(\begin{array}{ccc:ccc}1&5/3&2&0&0&1/3\\0&2/3&1&0&1&-2/3\\0&1/3&1&1&0&-1/3\end{array}\right)$$

$$\qquad\qquad\qquad\qquad\qquad\qquad\quad \boldsymbol{c}_3$$

$$\xrightarrow{\text{第 2 次消元}}\left(\begin{array}{ccc:ccc}1&0&-1/2&0&-5/2&2\\0&1&3/2&0&3/2&-1\\0&0&1/2&1&-1/2&0\end{array}\right)$$

$$\qquad\qquad\qquad\qquad\qquad\quad \boldsymbol{c}_2$$

$$\xrightarrow{\text{第 3 次消元}}\left(\begin{array}{ccc:ccc}1&0&0&1&-3&2\\0&1&0&-3&3&-1\\0&0&1&2&-1&0\end{array}\right)=(\boldsymbol{I}_n\,|\,\boldsymbol{A}^{-1}).$$

$$\qquad\qquad\quad \boldsymbol{c}_1$$

且 $\boldsymbol{m}_1=(m_{11},m_{21},m_{31})^{\mathrm{T}}=\boldsymbol{c}_3$, $\boldsymbol{m}_2=(m_{12},m_{22},m_{32})^{\mathrm{T}}=\boldsymbol{c}_2$, $\boldsymbol{m}_3=(m_{13},m_{23},m_{33})^{\mathrm{T}}=\boldsymbol{c}_1$.

为了节省内存单元,可不必将单位矩阵存放起来,$\boldsymbol{c}_3$ 存放在 $\boldsymbol{A}$ 的第 1 列位置,$\boldsymbol{c}_2$ 存放在 $\boldsymbol{A}$ 的第二列位置,$\boldsymbol{c}_1$ 存放在 $\boldsymbol{A}$ 的第 3 列位置,经消元计算,最后再调整一下列就可在 $\boldsymbol{A}$ 的位置得到 $\boldsymbol{A}^{-1}$. 注意第 k 步消元时,由 $\boldsymbol{A}$ 的第 k 列

$$\boldsymbol{a}_k=(a_{1k},\cdots,a_{kk},\cdots,a_{nk})^{\mathrm{T}}$$

计算

$$\boldsymbol{m}_k = \left(-\frac{a_{1k}}{a_{kk}}, \cdots, \frac{1}{a_{kk}}, \cdots, -\frac{a_{nk}}{a_{kk}}\right)^{\mathrm{T}},$$

且冲掉 $\boldsymbol{a}_k$.

最后在 $\boldsymbol{A}$ 位置如何调整列呢！事实上，我们在 $\boldsymbol{A}$ 位置最后得到矩阵 $\boldsymbol{PA}=\boldsymbol{A}_1$(其中 $\boldsymbol{P}$ 为排列阵)的逆矩阵 $\boldsymbol{A}_1^{-1}$，于是

$$\boldsymbol{A}^{-1} = \boldsymbol{A}_1^{-1}\boldsymbol{P}.$$

5.4 矩阵三角分解法

高斯消去法有很多变形，有的是高斯消去法的改进、改写，有的是用于某一类特殊性质矩阵的高斯消去法的简化.

5.4.1 直接三角分解法

将高斯消去法改写为紧凑形式，可以直接从矩阵 $\boldsymbol{A}$ 的元素得到计算 $\boldsymbol{L},\boldsymbol{U}$ 元素的递推公式，而不需任何中间步骤，这就是所谓直接三角分解法. 一旦实现了矩阵 $\boldsymbol{A}$ 的 LU 分解，那么求解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的问题就等价于求解两个三角形方程组

① $\boldsymbol{Ly}=\boldsymbol{b}$，求 $\boldsymbol{y}$；

② $\boldsymbol{Ux}=\boldsymbol{y}$，求 $\boldsymbol{x}$.

1. 不选主元的三角分解法

设 $\boldsymbol{A}$ 为非奇异矩阵，且有分解式

$$\boldsymbol{A} = \boldsymbol{LU},$$

其中 $\boldsymbol{L}$ 为单位下三角阵，$\boldsymbol{U}$ 为上三角阵，即

$$\boldsymbol{A} = \begin{pmatrix} 1 & & & \\ l_{21} & 1 & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & 1 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1} \\ & u_{22} & \cdots & u_{2n} \\ & & \ddots & \vdots \\ & & & u_{nn} \end{pmatrix}. \qquad (4.1)$$

下面说明 $\boldsymbol{L},\boldsymbol{U}$ 的元素可以由 n 步直接计算定出，其中第 r 步定出 $\boldsymbol{U}$ 的第 r 行和 $\boldsymbol{L}$ 的第 r 列元素．由(4.1)有：

$a_{1i}=u_{1i}\quad(i=1,2,\cdots,n)$，得 $\boldsymbol{U}$ 的第 1 行元素；

$a_{i1}=l_{i1}u_{11},l_{i1}=a_{i1}/u_{11}\quad(i=2,\cdots,n)$，得 $\boldsymbol{L}$ 的第 1 列元素．

设已经定出 $\boldsymbol{U}$ 的第 1 行到第 $r-1$ 行元素与 $\boldsymbol{L}$ 的第 1 列到第 $r-1$ 列元素．由(4.1)，利用矩阵乘法(注意当 $r<k$ 时，$l_{rk}=0$)，有

$$a_{ri}=\sum_{k=1}^{n}l_{rk}u_{ki}=\sum_{k=1}^{r-1}l_{rk}u_{ki}+u_{ri}.$$

故

$$u_{ri}=a_{ri}-\sum_{k=1}^{r-1}l_{rk}u_{ki}\quad(i=r,r+1,\cdots,n),$$

又由(4.1)有

$$a_{ir}=\sum_{k=1}^{n}l_{ik}u_{kr}=\sum_{k=1}^{r-1}l_{ik}u_{kr}+l_{ir}u_{rr}.$$

总结上述讨论，得到用直接三角分解法解 $\boldsymbol{Ax}=\boldsymbol{b}$(要求 $\boldsymbol{A}$ 的所有顺序主子式都不为零)的计算公式．

① $u_{1i}=a_{1i}(i=1,2,\cdots,n)$，$l_{i1}=a_{i1}/u_{11}(i=2,3,\cdots,n)$，

计算 $\boldsymbol{U}$ 的第 r 行，$\boldsymbol{L}$ 的第 r 列元素($r=2,3,\cdots,n$)．

$$\text{②}\ u_{ri}=a_{ri}-\sum_{k=1}^{r-1}l_{rk}u_{ki}\quad(i=r,r+1,\cdots,n);\tag{4.2}$$

$$\text{③}\ l_{ir}=\left(a_{ir}-\sum_{k=1}^{r-1}l_{ik}u_{kr}\right)/u_{rr}\quad(i=r+1,\cdots,n;\text{且 }r\neq n);\tag{4.3}$$

求解 $\boldsymbol{Ly}=\boldsymbol{b}$，$\boldsymbol{Ux}=\boldsymbol{y}$ 的计算公式；

$$\text{④}\begin{cases}y_1=b_1;\\ y_i=b_i-\sum_{k=1}^{i-1}l_{ik}y_k\quad(i=2,3,\cdots,n);\end{cases}\tag{4.4}$$

$$
⑤\begin{cases} x_n = y_n / u_{nn}; \\ x_i = \left(y_i - \sum_{k=i+1}^{n} u_{ik} x_k\right) / u_{ii} \end{cases} \tag{4.5}
$$

$$(i = n-1, n-2, \cdots, 1).$$

例 5 用直接三角分解法解

$$
\begin{pmatrix} 1 & 2 & 3 \\ 2 & 5 & 2 \\ 3 & 1 & 5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 14 \\ 18 \\ 20 \end{pmatrix}.
$$

解 用分解公式(4.2)—(4.3)计算得

$$
\boldsymbol{A} = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & -5 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & -4 \\ 0 & 0 & -24 \end{pmatrix} = \boldsymbol{LU}.
$$

求解

$$\boldsymbol{Ly} = (14, 18, 20)^{\mathrm{T}}, \text{ 得 } \boldsymbol{y} = (14, -10, -72)^{\mathrm{T}},$$

$$\boldsymbol{Ux} = (14, -10, -72)^{\mathrm{T}}, \text{ 得 } \boldsymbol{x} = (1, 2, 3)^{\mathrm{T}}.$$

由于在计算机实现时当 u_{ri} 计算好后 a_{ri} 就不用了，因此计算好 $\boldsymbol{L}, \boldsymbol{U}$ 的元素后就存放在 $\boldsymbol{A}$ 的相应位置. 例如

$$
\boldsymbol{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix} \to \begin{pmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ l_{21} & u_{22} & u_{23} & u_{24} \\ l_{31} & l_{32} & u_{33} & u_{34} \\ l_{41} & l_{42} & l_{43} & u_{44} \end{pmatrix}
$$

最后在存放 $\boldsymbol{A}$ 的数组中得到 $\boldsymbol{L}, \boldsymbol{U}$ 的元素.

由直接三角分解计算公式，需要计算形如 $\sum a_i b_i$ 的式子，可采用“双精度累加”，以提高精度.

直接分解法大约需要 $n^3/3$ 次乘除法，和高斯消去法计算量基本相同.

如果已经实现了 $\boldsymbol{A}=\boldsymbol{LU}$ 的分解计算，且 $\boldsymbol{L},\boldsymbol{U}$ 保存在 $\boldsymbol{A}$ 的相应位置，则用直接三角分解法解具有相同系数的方程组 $\boldsymbol{Ax}=(\boldsymbol{b}_1\boldsymbol{b}_2\cdots\ \boldsymbol{b}_m)$ 是相当方便的，每解一个方程组 $\boldsymbol{Ax}=\boldsymbol{b}_j$ 仅需要增加 n^2 次乘除法运算.

矩阵 $\boldsymbol{A}$ 的分解公式(4.2)，(4.3)又称为**杜利特尔**(Doolittle)**分解**.

2. 选主元的三角分解法

从直接三角分解公式可看出当 $u_{rr}=0$ 时计算将中断，或者当 u_{rr} 绝对值很小时，按分解公式计算可能引起舍入误差的累积. 但如果 $\boldsymbol{A}$ 非奇异，我们可通过交换 $\boldsymbol{A}$ 的行实现矩阵 $\boldsymbol{PA}$ 的 LU 分解，因此可采用与列主元消去法类似的方法(可以证明下述方法与列主元消去法等价)，将直接三角分解法修改为(部分)选主元的三角分解法.

设第 $r-1$ 步分解已完成，这时有

$$\boldsymbol{A}\to\begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1,r-1} & u_{1r} & \cdots & u_{1n} \\ l_{21} & u_{22} & \cdots & u_{2,r-1} & u_{2r} & \cdots & u_{2n} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ l_{r-1,1} & l_{r-1,2} & \cdots & u_{r-1,r-1} & u_{r-1,r} & \cdots & u_{r-1,n} \\ l_{r1} & l_{r2} & \cdots & l_{r,r-1} & a_{rr} & \cdots & a_{rn} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ l_{n1} & l_{n2} & \cdots & l_{n,r-1} & a_{nr} & \cdots & a_{nn} \end{pmatrix}.$$

第 r 步分解需用到(4.2)及(4.3)式，为了避免用小的数 u_{rr} 作除数，引进量

$$s_i=a_{ir}-\sum_{k=1}^{r-1}l_{ik}u_{kr}(i=r,r+1,\cdots,n).$$

于是有

$$u_{rr}=s_r,l_{ir}=s_i/s_r\quad(i=r+1,\cdots,n).$$

取$\max\limits_{r\leqslant i\leqslant n}|s_i|=|s_{i_r}|$，交换 A 的 r 行与 i_r 行元素，将 s_{i_r} 调到(r,r)位置(将(i,j)位置的新元素仍记为 l_{ij} 及 a_{ij})，于是有$|l_{ir}|\leqslant 1(i=r+1,\cdots,n)$. 由此再进行第 r 步分解计算.

算法 4(选主元的三角分解法) 设 $\boldsymbol{Ax}=\boldsymbol{b}$，其中 $\boldsymbol{A}$ 为非奇异矩阵. 本算法采用选主元的三角分解法，用 $\boldsymbol{PA}=\boldsymbol{I}_{n-1,i_{n-1}}\cdots\boldsymbol{I}_{1,i_1}\boldsymbol{A}$ 的三角分解冲掉$\boldsymbol{A}$，用整型数组 Ip(n)记录主行，解 $\boldsymbol{x}$ 存放在 $\boldsymbol{b}$ 内.

1. 对于 $r=1,2,\cdots,n$

(1) 计算 s_i

$$a_{ir}\leftarrow s_i=a_{ir}-\sum_{k=1}^{r-1}l_{ik}u_{kr}\quad(i=r,r+1,\cdots,n)$$

(2) 选主元 $|s_{i_r}|=\max\limits_{r\leqslant i\leqslant n}|s_i|$, $\mathrm{Ip}(r)\leftarrow i_r$

(3) 交换 $\boldsymbol{A}$ 的 r 行与 i_r 行元素

$$a_{ri}\longleftrightarrow a_{i_r,i}\quad(i=1,2,\cdots,n)$$

(4) 计算 $\boldsymbol{U}$ 的第 r 行元素，$\boldsymbol{L}$ 的第 r 列元素

$$a_{rr}=u_{rr}=s_r$$

$$a_{ir}\leftarrow l_{ir}=s_i/u_{rr}=a_{ir}/a_{rr}\quad(i=r+1,\cdots,n,\text{且 }r\neq n)$$

$$a_{ri}\leftarrow u_{ri}=a_{ri}-\sum_{k=1}^{r-1}l_{rk}u_{ki}\quad(i=r+1,\cdots,n,\text{且 }r\neq n)$$

(这时有$|l_{ir}|\leqslant 1$).

上述计算过程完成后就实现了 $\boldsymbol{PA}$ 的 LU 分解，且 $\boldsymbol{U}$ 保存在 $\boldsymbol{A}$ 的上三角部分，$\boldsymbol{L}$ 保存在 $\boldsymbol{A}$ 的下三角部分，排列阵 $\boldsymbol{P}$ 由 Ip(n)最后记录可知.

求解 $\boldsymbol{Ly}=\boldsymbol{Pb}$ 及 $\boldsymbol{Ux}=\boldsymbol{y}$.

2. 对于 $i=1,\cdots,n-1$

(1) $t\leftarrow\mathrm{Ip}(i)$

(2) 如果 $i=t$ 则转(3)

$b_i\longleftrightarrow b_t$

(3)(继续循环)

3. $b_i \leftarrow b_i - \sum_{k=1}^{i-1} l_{ik} b_k \quad (i = 2,3,\cdots,n)$

4. $b_n \leftarrow b_n/u_{nn}, b_i \leftarrow \left(b_i - \sum_{k=i+1}^{n} u_{ik} b_k\right)/u_{ii}$

$(i=n-1,\cdots,1)$

利用算法 4 的结果(实现 $\boldsymbol{PA}=\boldsymbol{LU}$ 三角分解),则可以计算 $\boldsymbol{A}$ 的逆矩阵

$$\boldsymbol{A}^{-1} = \boldsymbol{U}^{-1}\boldsymbol{L}^{-1}\boldsymbol{P}.$$

利用 $\boldsymbol{PA}$ 的三角分解计算 $\boldsymbol{A}^{-1}$ 步骤:

(1) 计算上三角矩阵的逆阵 $\boldsymbol{U}^{-1}$;

(2) 计算 $\boldsymbol{U}^{-1}\boldsymbol{L}^{-1}$;

(3) 交换 $\boldsymbol{U}^{-1}\boldsymbol{L}^{-1}$ 列(利用 Ip(n)最后记录).

上述方法求 $\boldsymbol{A}^{-1}$ 大约需要 n^3 次乘法运算.

5.4.2 平方根法

应用有限元法解结构力学问题时,最后归结为求解线性方程组,系数矩阵大多具有对称正定性质. 所谓平方根法,就是利用对称正定矩阵的三角分解而得到的求解对称正定方程组的一种有效方法,目前在计算机上广泛应用平方根法解此类方程组.

设 $\boldsymbol{A}$ 为对称矩阵,且 $\boldsymbol{A}$ 的所有顺序主子式均不为零,由本章定理 7 知,$\boldsymbol{A}$ 可唯一分解为如(4.1)的形式.

为了利用 $\boldsymbol{A}$ 的对称性,将 $\boldsymbol{U}$ 再分解为

$$\boldsymbol{U} = \begin{pmatrix} u_{11} & & & \\ & u_{22} & & \\ & & \ddots & \\ & & & u_{nn} \end{pmatrix} \begin{pmatrix} 1 & \frac{u_{12}}{u_{11}} & \cdots & \cdots & \frac{u_{1n}}{u_{11}} \\ & 1 & \frac{u_{23}}{u_{22}} & \cdots & \frac{u_{2n}}{u_{22}} \\ & & \ddots & & \vdots \\ & & & & 1 \end{pmatrix} = \boldsymbol{DU}_0,$$

其中 $\boldsymbol{D}$ 为对角阵，$\boldsymbol{U}_0$ 为单位上三角阵. 于是

$$\boldsymbol{A}=\boldsymbol{LU}=\boldsymbol{LDU}_0. \tag{4.6}$$

又

$$\boldsymbol{A}=\boldsymbol{A}^{\mathrm{T}}=\boldsymbol{U}_0^{\mathrm{T}}(\boldsymbol{DL}^{\mathrm{T}}),$$

由分解的唯一性即得

$$\boldsymbol{U}_0^{\mathrm{T}}=\boldsymbol{L}.$$

代入(4.6)得到对称矩阵 $\boldsymbol{A}$ 的分解式 $\boldsymbol{A}=\boldsymbol{LDL}^{\mathrm{T}}$. 总结上述讨论有

定理 10(对称阵的三角分解定理) 设 $\boldsymbol{A}$ 为 n 阶对称阵，且 $\boldsymbol{A}$ 的所有顺序主子式均不为零，则 $\boldsymbol{A}$ 可唯一分解为

$$\boldsymbol{A}=\boldsymbol{LDL}^{\mathrm{T}},$$

其中 $\boldsymbol{L}$ 为单位下三角阵，$\boldsymbol{D}$ 为对角阵.

现设 $\boldsymbol{A}$ 为对称正定矩阵. 首先说明 $\boldsymbol{A}$ 的分解式 $\boldsymbol{A}=\boldsymbol{LDL}^{\mathrm{T}}$ 中 $\boldsymbol{D}$ 的对角元素 d_i 均为正数.

事实上，由 $\boldsymbol{A}$ 的对称正定性，5.2 节中的推论成立，即

$$d_1=D_1>0,\ d_i=D_i/D_{i-1}>0 \quad (i=2,3,\cdots,n).$$

于是

$$\boldsymbol{D}=\begin{pmatrix} d_1 & & \\ & \ddots & \\ & & d_n \end{pmatrix}=\begin{pmatrix} \sqrt{d_1} & & \\ & \ddots & \\ & & \sqrt{d_n} \end{pmatrix}\begin{pmatrix} \sqrt{d_1} & & \\ & \ddots & \\ & & \sqrt{d_n} \end{pmatrix}$$

$$=\boldsymbol{D}^{\frac{1}{2}}\boldsymbol{D}^{\frac{1}{2}},$$

由定理 6 得到

$$\begin{aligned}\boldsymbol{A}&=\boldsymbol{LDL}^{\mathrm{T}}=\boldsymbol{LD}^{\frac{1}{2}}\boldsymbol{D}^{\frac{1}{2}}\boldsymbol{L}^{\mathrm{T}}=(\boldsymbol{LD}^{\frac{1}{2}})(\boldsymbol{LD}^{\frac{1}{2}})^{\mathrm{T}}\\&=\boldsymbol{L}_1\boldsymbol{L}_1^{\mathrm{T}},\end{aligned}$$

其中 $\boldsymbol{L}_1=\boldsymbol{LD}^{\frac{1}{2}}$ 为下三角矩阵.

定理 11(对称正定矩阵的三角分解或 Cholesky 分解) 如果 $\boldsymbol{A}$ 为 n 阶对称正定矩阵，则存在一个实的非奇异下三角阵 $\boldsymbol{L}$ 使

$\boldsymbol{A}=\boldsymbol{L}\boldsymbol{L}^{\mathrm{T}}$，当限定 $\boldsymbol{L}$ 的对角元素为正时，这种分解是唯一的.

下面我们用直接分解方法来确定计算 $\boldsymbol{L}$ 元素的递推公式. 因为

$$\boldsymbol{A}=\begin{pmatrix} l_{11} & & & \\ l_{21} & l_{22} & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{pmatrix}\begin{pmatrix} l_{11} & l_{21} & \cdots & l_{n1} \\ & l_{22} & \cdots & l_{n2} \\ & & \ddots & \vdots \\ & & & l_{nn} \end{pmatrix},$$

其中 $l_{ii}>0(i=1,2,\cdots,n)$. 由矩阵乘法及 $l_{jk}=0$（当 $j<k$ 时），得

$$a_{ij}=\sum_{k=1}^{n} l_{ik}l_{jk}=\sum_{k=1}^{j-1} l_{ik}l_{jk}+l_{jj}l_{ij},$$

于是得到解对称正定方程组 $\boldsymbol{Ax}=\boldsymbol{b}$ 的平方根法计算公式：

对于　$j=1,2,\cdots,n$

$$1.\ l_{jj}=\left(a_{jj}-\sum_{k=1}^{j-1} l_{jk}^{2}\right)^{\frac{1}{2}}, \tag{4.7}$$

$$2.\ l_{ij}=\left(a_{ij}-\sum_{k=1}^{j-1} l_{ik}l_{jk}\right)/l_{jj}\quad (i=j+1,\cdots,n);$$

求解 $\boldsymbol{Ax}=\boldsymbol{b}$，即求解两个三角形方程组：

(1) $\boldsymbol{Ly}=\boldsymbol{b}$，求 $\boldsymbol{y}$；　　(2) $\boldsymbol{L}^{\mathrm{T}}\boldsymbol{x}=\boldsymbol{y}$，求 $\boldsymbol{x}$.

$$3.\ y_i=\left(b_i-\sum_{k=1}^{i-1} l_{ik}y_k\right)/l_{ii}\quad (i=1,2,\cdots,n). \tag{4.8}$$

$$4.\ x_i=\left(b_i-\sum_{k=i+1}^{n} l_{ki}x_k\right)/l_{ii}\quad (i=n,n-1,\cdots,1).$$

由计算公式 1 知

$$a_{jj}=\sum_{k=1}^{j} l_{jk}^{2}\quad (j=1,2,\cdots,n),$$

所以

$$l_{jk}^{2}\leqslant a_{jj}\leqslant \max_{1\leqslant j\leqslant n}\{a_{jj}\},$$

于是

$$\max_{j,k}\{l_{jk}^2\} \leqslant \max_{1\leqslant j\leqslant n}\{a_{jj}\}.$$

上面分析说明，分解过程中元素 l_{jk} 的数量级不会增长且对角元素 l_{jj} 恒为正数. 于是不选主元素的平方根法是一个数值稳定的方法.

当求出 $\boldsymbol{L}$ 的第 j 列元素时，$\boldsymbol{L}^{\mathrm{T}}$ 的第 j 行元素亦算出. 所以平方根法约需 $n^3/6$ 次乘除法，大约为一般直接 $\boldsymbol{LU}$ 分解法计算量的一半.

由于 $\boldsymbol{A}$ 为对称阵，因此在计算机实现时只需存储 $\boldsymbol{A}$ 的下三角部分，共需要存储 $n(n+1)/2$ 个元素，可用一维数组存放，即

$$A(n\cdot(n+1)/2)=\{a_{11},a_{21},a_{22},\cdots,a_{n1},a_{n2},\cdots,a_{nn}\}.$$

矩阵元素 a_{ij} 一维数组的表示为 $A(i\cdot(i-1)/2+j)$，$\boldsymbol{L}$ 的元素存放在 A 的相应位置.

由公式(4.7)看出，用平方根法解对称正定方程组时，计算 $\boldsymbol{L}$ 的元素 l_{ii} 需要用到开方运算. 为了避免开方，我们下面用定理 10 的分解式

$$\boldsymbol{A}=\boldsymbol{LDL}^{\mathrm{T}},$$

即

$$\boldsymbol{A}=\begin{pmatrix}1 & & & \\ l_{21} & 1 & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & 1\end{pmatrix}\begin{pmatrix}d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_n\end{pmatrix}\begin{pmatrix}1 & l_{21} & \cdots & l_{n1}\\ & 1 & \cdots & l_{n2}\\ & & \ddots & \vdots \\ & & & 1\end{pmatrix}.$$

由矩阵乘法，并注意 $l_{jj}=1$，$l_{jk}=0(j<k)$，得

$$\begin{aligned}a_{ij}&=\sum_{k=1}^{n}(\boldsymbol{LD})_{ik}(\boldsymbol{L}^{\mathrm{T}})_{kj}=\sum_{k=1}^{n}l_{ik}d_kl_{jk}\\&=\sum_{k=1}^{j-1}l_{ik}d_kl_{jk}+l_{ij}d_jl_{jj}.\end{aligned}$$

于是得到计算 $\boldsymbol{L}$ 的元素及 $\boldsymbol{D}$ 的对角元素公式：

对于 $i=1,2,\cdots,n$.

1. $l_{ij}=(a_{ij}-\sum_{k=1}^{j-1}l_{ik}d_kl_{jk})/d_j \quad (j=1,2,\cdots,i-1)$; (4.9)

2. $d_i=a_{ii}-\sum_{k=1}^{i-1}l_{ik}^2d_k$.

为了避免重复计算,我们引进

$$t_{ij}=l_{ij}d_j,$$

由(4.9)得到按行计算 $\boldsymbol{L},\boldsymbol{T}$ 元素的公式:

$$d_1=a_{11}$$

对于 $i=2,3,\cdots,n$.

1. $t_{ij}=a_{ij}-\sum_{k=1}^{j-1}t_{ik}l_{jk}, \quad (j=1,2,\cdots,i-1)$;

2. $l_{ij}=t_{ij}/d_j \quad (j=1,2,\cdots,i-1)$;

3. $d_i=a_{ii}-\sum_{k=1}^{i-1}t_{ik}l_{ik}$. (4.10)

计算出 $\boldsymbol{T}=\boldsymbol{LD}$ 的第 i 行元素 $t_{ij}(j=1,2,\cdots,i-1)$后,存放在 $\boldsymbol{A}$ 的第 i 行相应位置,然后再计算 $\boldsymbol{L}$ 的第 i 行元素,存放在 $\boldsymbol{A}$ 的第 i 行. $\boldsymbol{D}$ 的对角元素存放在 $\boldsymbol{A}$ 的相应位置. 例如

$$\boldsymbol{A}=\begin{pmatrix} a_{11} & & & \\ a_{21} & a_{22} & \text{对称} & \\ a_{31} & a_{32} & a_{33} & \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix} \to \begin{pmatrix} d_1 & & & \\ l_{21} & d_2 & & \\ l_{31} & l_{32} & d_3 & \\ t_{41} & t_{42} & t_{43} & a_{44} \end{pmatrix}$$

$$\to \begin{pmatrix} d_1 & & & \\ l_{21} & d_2 & & \\ l_{31} & l_{32} & d_3 & \\ l_{41} & l_{42} & l_{43} & d_4 \end{pmatrix}.$$

对称正定矩阵 $\boldsymbol{A}$ 按 $\boldsymbol{LDL}^{\mathrm{T}}$ 分解和按 $\boldsymbol{LL}^{\mathrm{T}}$ 分解计算量差不多,

但 $\boldsymbol{LDL}^{\mathrm{T}}$ 分解不需要开方计算.

求解 $\boldsymbol{Ly}=\boldsymbol{b}$, $\boldsymbol{DL}^{\mathrm{T}}\boldsymbol{x}=\boldsymbol{y}$ 计算公式

$$4.\begin{cases} y_1 = b_1; \\ y_i = b_i - \sum\limits_{k=1}^{i-1} l_{ik} y_k \quad (i = 2, \cdots, n). \end{cases} \tag{4.11}$$

$$5.\begin{cases} x_n = y_n / d_n; \\ x_i = y_i / d_i - \sum\limits_{k=i+1}^{n} l_{ki} x_k \quad (i = n-1, \cdots, 2, 1). \end{cases}$$

计算公式(4.10),(4.11)称为**改进的平方根法**.

5.4.3 追赶法

在一些实际问题中,例如解常微分方程边值问题,解热传导方程以及船体数学放样中建立三次样条函数等,都会要求解系数矩阵为对角占优的三对角线方程组

$$\begin{pmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & a_n & b_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_{n-1} \\ f_n \end{pmatrix}, \tag{4.12}$$

简记为 $\boldsymbol{Ax}=\boldsymbol{f}$. 其中,当 $|i-j|>1$ 时, $a_{ij}=0$, 且:

(a) $|b_1|>|c_1|>0$;

(b) $|b_i|\geqslant|a_i|+|c_i|$, $a_i, c_i\neq 0 \quad (i=2,3,\cdots,n-1)$;

(c) $|b_n|>|a_n|>0$.

我们利用矩阵的直接三角分解法来推导解三对角线方程组(4.12)的计算公式. 由系数阵 $\boldsymbol{A}$ 的特点,可以将 $\boldsymbol{A}$ 分解为两个三角阵的乘积,即

$$\boldsymbol{A} = \boldsymbol{LU},$$

其中 $\boldsymbol{L}$ 为下三角矩阵,$\boldsymbol{U}$ 为单位上三角矩阵. 下面我们来说明这种分解是可能的. 设

$$\boldsymbol{A}=\begin{pmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & a_n & b_n \end{pmatrix} \tag{4.13}$$

$$=\begin{pmatrix} \alpha_1 & & & \\ r_2 & \alpha_2 & & \\ & \ddots & \ddots & \\ & & r_n & \alpha_n \end{pmatrix}\begin{pmatrix} 1 & \beta_1 & & \\ & 1 & \ddots & \\ & & \ddots & \beta_{n-1} \\ & & & 1 \end{pmatrix},$$

其中 α_i,β_i,r_i 为待定系数. 比较(4.13)两边即得

$$\left.\begin{aligned} & b_1=\alpha_1,c_1=\alpha_1\beta_1, \\ & a_i=r_i,b_i=r_i\beta_{i-1}+\alpha_i \quad (i=2,\cdots,n), \\ & c_i=\alpha_i\beta_i \quad (i=2,3,\cdots,n-1). \end{aligned}\right\} \tag{4.14}$$

由 $\alpha_1=b_1\neq0$, $|b_1|>|c_1|>0$, $\beta_1=c_1/b_1$,得 $0<|\beta_1|<1$. 下面我们用归纳法证明

$$|\alpha_i|>|c_i|\neq0 \quad (i=1,2,\cdots,n-1), \tag{4.15}$$

即 $0<|\beta_i|<1$. 从而由(4.14)可求出 β_i.

(4.15)对 $i=1$ 是成立的. 现设(4.15)对 $i-1$ 成立,求证对 i 亦成立.

由归纳法假设 $0<|\beta_{i-1}|<1$, 又由(4.15)及 $\boldsymbol{A}$ 的假设条件有

$$\begin{aligned} |\alpha_i| &= |b_i-a_i\beta_{i-1}|\geqslant|b_i|-|a_i\beta_{i-1}| \\ &>|b_i|-|a_i|\geqslant|c_i|\neq0, \end{aligned}$$

也就是 $0<|\beta_i|<1$. 由(4.14)得到

$$\alpha_i=b_i-a_i\beta_{i-1} \quad (i=2,\cdots,n);$$

$$\beta_i = c_i/(b_i - a_i\beta_{i-1}) \quad (i = 2,3,\cdots,n-1).$$

这就是说，由 $\boldsymbol{A}$ 的假设条件，我们完全确定了 $\{a_i\}$，$\{\beta_i\}$，$\{r_i\}$，实现了 $\boldsymbol{A}$ 的 $\boldsymbol{LU}$ 分解.

求解 $\boldsymbol{Ax}=\boldsymbol{f}$ 等价于解两个三角形方程组

(1) $\boldsymbol{Ly}=\boldsymbol{f}$，求 $\boldsymbol{y}$；(2) $\boldsymbol{Ux}=\boldsymbol{y}$，求 $\boldsymbol{x}$.

从而得到解三对角线方程组的**追赶法公式**：

1. 计算 $\{\beta_i\}$ 的递推公式

$\beta_1 = c_1/b_1$,

$\beta_i = c_i/(b_i - a_i\beta_{i-1}) \quad (i=2,3,\cdots,n-1)$;

2. 解 $\boldsymbol{Ly}=\boldsymbol{f}$

$y_1 = f_1/b_1$,

$y_i = (f_i - a_i y_{i-1})/(b_i - a_i\beta_{i-1}) \quad (i=2,3,\cdots,n)$;

3. 解 $\boldsymbol{Ux}=\boldsymbol{y}$

$x_n = y_n$,

$x_i = y_i - \beta_i x_{i+1} \quad (i=n-1,n-2,\cdots,2,1)$.

我们将计算系数 $\beta_1 \to \beta_2 \to \cdots \to \beta_{n-1}$ 及 $y_1 \to y_2 \to \cdots \to y_n$ 的过程称为追的过程，将计算方程组的解 $x_n \to x_{n-1} \to \cdots \to x_1$ 的过程称为赶的过程.

总结上述讨论有

定理 12 设有三对角线方程组 $\boldsymbol{Ax}=\boldsymbol{f}$，其中 $\boldsymbol{A}$ 满足条件 (a)，(b)，(c)，则 $\boldsymbol{A}$ 为非奇异矩阵且追赶法计算公式中的 $\{\alpha_i\}$，$\{\beta_i\}$ 满足：

1° $0<|\beta_i|<1 \quad (i=1,2,\cdots,n-1)$;

2° $0<|c_i| \leqslant |b_i| - |a_i| < |\alpha_i| < |b_i| + |a_i| \ (i=2,3,\cdots,n-1)$;

$0<|b_n| - |a_n| < |\alpha_n| < |b_n| + |a_n|$.

追赶法公式实际上就是把高斯消去法用到求解三对角线方程

组上去的结果．这时由于 $\boldsymbol{A}$ 特别简单，因此使得求解的计算公式非常简单，而且计算量仅为 $5n-4$ 次乘除法，而另外增加解一个方程组 $\boldsymbol{Ax}=\boldsymbol{f}_2$ 仅增加 $3n-2$ 次乘除运算．易见追赶法的计算量是比较小的．

由定理 12 的 1°，2°说明追赶法计算公式中不会出现中间结果数量级的巨大增长和舍入误差的严重累积．

在计算机实现时我们只需用三个一维数组分别存储 $\boldsymbol{A}$ 的三条线元素 $\{a_i\}$，$\{b_i\}$，$\{c_i\}$，此外还需要用两组工作单元保存 $\{\beta_i\}$，$\{y_i\}$ 或 $\{x_i\}$．

5.5 向量和矩阵的范数

为了研究线性方程组近似解的误差估计和迭代法的收敛性，我们需要对 $\mathbf{R}^n$（n 维向量空间）中向量（或 $\mathbf{R}^{n\times n}$ 中矩阵）的“大小”引进某种度量——向量（或矩阵）范数的概念．向量范数概念是三维欧氏空间中向量长度概念的推广，在数值分析中起着重要作用．

首先将向量长度概念推广到 $\mathbf{R}^n$（或 $\mathbf{C}^n$）中．

定义 1 设 $\boldsymbol{x}=(x_1,x_2,\cdots,x_n)^{\mathrm{T}}$，$\boldsymbol{y}=(y_1,y_2,\cdots,y_n)^{\mathrm{T}}\in\mathbf{R}^n$（或 $\mathbf{C}^n$）．将实数 $(\boldsymbol{x},\boldsymbol{y})=\boldsymbol{y}^{\mathrm{T}}\boldsymbol{x}=\sum\limits_{i=1}^{n}x_iy_i$（或复数 $(\boldsymbol{x},\boldsymbol{y})=\boldsymbol{y}^{\mathrm{H}}\boldsymbol{x}=\sum\limits_{i=1}^{n}x_i\bar{y}_i$ 称为向量 $\boldsymbol{x},\boldsymbol{y}$ 的**数量积**．将非负实数 $\|\boldsymbol{x}\|_2=(\boldsymbol{x},\boldsymbol{x})^{\frac{1}{2}}=(\sum\limits_{i=1}^{n}x_i^2)^{\frac{1}{2}}$ 或 $\|\boldsymbol{x}\|_2=(\boldsymbol{x},\boldsymbol{x})^{\frac{1}{2}}=\left(\sum\limits_{i=1}^{n}|x_i|^2\right)^{\frac{1}{2}}$ 称为向量 $\boldsymbol{x}$ 的**欧氏范数**．

下述定理可在线性代数书中找到．

定理 13 设 $\boldsymbol{x},\boldsymbol{y}\in\mathbf{R}^n$(或 $\mathbf{C}^n$),则

1. $(\boldsymbol{x},\boldsymbol{x})=0$,当且仅当 $\boldsymbol{x}=\mathbf{0}$ 时成立;

2. $(\alpha\boldsymbol{x},\boldsymbol{y})=\alpha(\boldsymbol{x},\boldsymbol{y})$. α 为实数(或$(\boldsymbol{x},\alpha\boldsymbol{y})=\bar{\alpha}(\boldsymbol{x},\boldsymbol{y})$,$\alpha$ 为复数);

3. $(\boldsymbol{x},\boldsymbol{y})=(\boldsymbol{y},\boldsymbol{x})$(或$(\boldsymbol{x},\boldsymbol{y})=\overline{(\boldsymbol{y},\boldsymbol{x})}$);

4. $(\boldsymbol{x}_1+\boldsymbol{x}_2,\ \boldsymbol{y})=(\boldsymbol{x}_1,\boldsymbol{y})+(\boldsymbol{x}_2,\boldsymbol{y})$;

5. (Cauchy-Schwarz 不等式)

$$|(\boldsymbol{x},\boldsymbol{y})|\leqslant\|\boldsymbol{x}\|_2\cdot\|\boldsymbol{y}\|_2,$$

等式当且仅当 $\boldsymbol{x}$ 与 $\boldsymbol{y}$ 线性相关时成立;

6. 三角不等式

$$\|\boldsymbol{x}+\boldsymbol{y}\|_2\leqslant\|\boldsymbol{x}\|_2+\|\boldsymbol{y}\|_2.$$

我们还可以用其他办法来度量 $\mathbf{R}^n$ 中向量的"大小". 例如,对于 $\boldsymbol{x}=(x_1,x_2)^{\mathrm{T}}\in\mathbf{R}^2$,可以用一个 $\boldsymbol{x}$ 的函数 $N(\boldsymbol{x})=\max\limits_{i=1,2}|x_i|$ 来度量 $\boldsymbol{x}$ 的"大小",而且这种度量 $\boldsymbol{x}$"大小"的方法计算起来比欧氏范数方便. 在许多应用中,对度量向量 $\boldsymbol{x}$"大小"的函数 $N(\boldsymbol{x})$都要求是正定的、齐次的且满足三角不等式. 下面我们给出向量范数的一般定义.

定义 2(向量的范数) 如果向量 $\boldsymbol{x}\in\mathbf{R}^n$(或 $\mathbf{C}^n$)的某个实值函数 $N(\boldsymbol{x})=\|\boldsymbol{x}\|$,满足条件:

(1) $\|\boldsymbol{x}\|\geqslant 0$($\|\boldsymbol{x}\|=0$ 当且仅当 $\boldsymbol{x}=\mathbf{0}$) (正定条件),

(2) $\|\alpha\boldsymbol{x}\|=|\alpha|\,\|\boldsymbol{x}\|$,$\forall\alpha\in\mathbf{R}$(或 $\alpha\in\mathbf{C}$), (5.1)

(3) $\|\boldsymbol{x}+\boldsymbol{y}\|\leqslant\|\boldsymbol{x}\|+\|\boldsymbol{y}\|$(三角不等式),

则称 $N(\boldsymbol{x})$是 $\mathbf{R}^n$(或 $\mathbf{C}^n$)上的一个向量范数(或模). 由(3)可推出不等式

(4) $\big|\,\|\boldsymbol{x}\|-\|\boldsymbol{y}\|\,\big|\leqslant\|\boldsymbol{x}-\boldsymbol{y}\|$. (5.2)

下面我们给出几种常用的向量范数.

1. 向量的∞-范数(最大范数):

$$\| \boldsymbol{x} \|_\infty = \max_{1 \leqslant i \leqslant n} | x_i |.$$

容易验证这样定义的向量 $\boldsymbol{x}$ 的函数 $N(\boldsymbol{x}) = \| \boldsymbol{x} \|_\infty$ 满足向量范数的三个条件.

2. 向量的 1-范数：

$$\| \boldsymbol{x} \|_1 = \sum_{i=1}^{n} | x_i |.$$

同样可证 $N(\boldsymbol{x}) = \| \boldsymbol{x} \|_1$ 是 $\mathbf{R}^n$ 上的一个向量范数.

3. 向量的 2-范数：

$$\| \boldsymbol{x} \|_2 = (\boldsymbol{x}, \boldsymbol{x})^{\frac{1}{2}} = (\sum_{i=1}^{n} x_i^2)^{\frac{1}{2}}.$$

由定理 13 知 $N(\boldsymbol{x}) = \| \boldsymbol{x} \|_2$ 是 $\mathbf{R}^n$ 上一个向量范数，称为向量 $\boldsymbol{x}$ 的欧氏范数.

4. 向量的 p-范数：

$$\| \boldsymbol{x} \|_p = (\sum_{i=1}^{n} | x_i |^p)^{1/p},$$

其中 $p \in [1, \infty)$，可以证明向量函数 $N(\boldsymbol{x}) \equiv \| \boldsymbol{x} \|_p$ 是 $\mathbf{R}^n$ 上向量的范数且容易说明上述三种范数是 p-范数的特殊情况（$\| \boldsymbol{x} \|_\infty = \lim_{p \to \infty} \| x \|_p$）.

例 6　计算向量 $\boldsymbol{x} = (1, -2, 3)^{\mathrm{T}}$ 的各种范数.

解　$\| \boldsymbol{x} \|_1 = 6$，$\| \boldsymbol{x} \|_\infty = 3$，$\| \boldsymbol{x} \|_2 = \sqrt{14}$.

定义 3　设 $\{\boldsymbol{x}^{(k)}\}$ 为 $\mathbf{R}^n$ 中一向量序列，$\boldsymbol{x}^* \in \mathbf{R}^n$，记 $\boldsymbol{x}^{(k)} = (x_1^{(k)}, x_2^{(k)}, \cdots, x_n^{(k)})^{\mathrm{T}}$，$\boldsymbol{x}^* = (x_1^*, \cdots, x_n^*)^{\mathrm{T}}$. 如果 $\lim_{k \to \infty} x_i^{(k)} = x_i^*$ $(i = 1, 2, \cdots, n)$，则称 $\boldsymbol{x}^{(k)}$ 收敛于向量 $\boldsymbol{x}^*$，记为

$$\lim_{k \to \infty} \boldsymbol{x}^{(k)} = \boldsymbol{x}^*.$$

定理 14（$N(\boldsymbol{x})$ 的连续性）　设非负函数 $N(\boldsymbol{x}) = \| \boldsymbol{x} \|$ 为 $\mathbf{R}^n$ 上任一向量范数，则 $N(\boldsymbol{x})$ 是 $\boldsymbol{x}$ 的分量 $x_1, x_2, \cdots, x_n$ 的连续函数.

证明 设 $\boldsymbol{x}=\sum_{i=1}^{n}x_i\boldsymbol{e}_i$，$\boldsymbol{y}=\sum_{i=1}^{n}y_i\boldsymbol{e}_i$，其中 $\boldsymbol{e}_i=(0,\cdots,1,0,\cdots,0)^{\mathrm{T}}$.

只须证明当 $\boldsymbol{x}\to\boldsymbol{y}$ 时 $N(\boldsymbol{x})\to N(\boldsymbol{y})$ 即成. 事实上

$$\begin{aligned}|N(\boldsymbol{x})-N(\boldsymbol{y})| &= |\,\|\boldsymbol{x}\|-\|\boldsymbol{y}\|\,| \leqslant \|\boldsymbol{x}-\boldsymbol{y}\| \\ &= \left\|\sum_{i=1}^{n}(x_i-y_i)\boldsymbol{e}_i\right\| \\ &\leqslant \sum_{i=1}^{n}|x_i-y_i|\,\|\boldsymbol{e}_i\| \\ &\leqslant \|\boldsymbol{x}-\boldsymbol{y}\|_\infty\sum_{i=1}^{n}\|\boldsymbol{e}_i\|,\end{aligned}$$

即 $|N(\boldsymbol{x})-N(\boldsymbol{y})|\leqslant c\|\boldsymbol{x}-\boldsymbol{y}\|_\infty\to 0$ （当 $\boldsymbol{x}\to\boldsymbol{y}$ 时），

其中

$$c=\sum_{i=1}^{n}\|\boldsymbol{e}_i\|.$$

定理 15（向量范数的等价性） 设 $\|\boldsymbol{x}\|_s$，$\|\boldsymbol{x}\|_t$ 为 $\mathbf{R}^n$ 上向量的任意两种范数，则存在常数 $c_1,c_2>0$，使得对一切 $\boldsymbol{x}\in\mathbf{R}^n$ 有

$$c_1\|\boldsymbol{x}\|_s\leqslant\|\boldsymbol{x}\|_t\leqslant c_2\|\boldsymbol{x}\|_s.$$

证明 只要就 $\|\boldsymbol{x}\|_s=\|\boldsymbol{x}\|_\infty$ 证明上式成立即可，即证明存在常数 $c_1,c_2>0$，使

$$c_1\leqslant\frac{\|\boldsymbol{x}\|_t}{\|\boldsymbol{x}\|_\infty}\leqslant c_2\text{，对一切 }\boldsymbol{x}\in\mathbf{R}^n\text{ 且 }\boldsymbol{x}\neq\mathbf{0}.$$

考虑泛函

$$f(\boldsymbol{x})=\|\boldsymbol{x}\|_t\geqslant 0,\ \boldsymbol{x}\in\mathbf{R}^n.$$

记 $S=\{\boldsymbol{x}\,|\,\|\boldsymbol{x}\|_\infty=1,\ \boldsymbol{x}\in\mathbf{R}^n\}$，则 S 是一个有界闭集. 由于 $f(\boldsymbol{x})$ 为 S 上的连续函数，所以 $f(\boldsymbol{x})$ 于 S 上达到最大最小值，即存在 $\boldsymbol{x}',\boldsymbol{x}''\in S$ 使得

$$f(\boldsymbol{x}')=\min_{\boldsymbol{x}\in S}f(\boldsymbol{x})=c_1,\ f(\boldsymbol{x}'')=\max_{\boldsymbol{x}\in S}f(\boldsymbol{x})=c_2.$$

设 $x \in \mathbf{R}^n$ 且 $x \neq \mathbf{0}$，则 $\frac{x}{\|x\|_\infty} \in S$，从而有

$$c_1 \leqslant f\left(\frac{x}{\|x\|_\infty}\right) \leqslant c_2, \tag{5.3}$$

显然 $c_1, c_2 > 0$，上式为

$$c_1 \leqslant \left\|\frac{x}{\|x\|_\infty}\right\|_t \leqslant c_2,$$

即

$$c_1\|x\|_\infty \leqslant \|x\|_t \leqslant c_2\|x\|_\infty, \text{ 对一切 } x \in \mathbf{R}^n.$$

注意，定理 15 不能推广到无穷维空间. 由定理 15 可得到结论：如果在一种范数意义下向量序列收敛时，则在任何一种范数意义下该向量序列均收敛.

定理 16 $\lim\limits_{k\to\infty} x^{(k)} = x^* \Leftrightarrow \lim\limits_{k\to\infty}\|x^{(k)} - x^*\| = 0$，其中 $\|\cdot\|$ 为向量的任一种范数.

证明 显然，$\lim\limits_{k\to\infty} x^{(k)} = x^* \Leftrightarrow \lim\limits_{k\to\infty}\|x^{(k)} - x^*\|_\infty = 0$，而对于 $\mathbf{R}^n$ 上任一种范数 $\|\cdot\|$，由定理 15，存在常数 $c_1, c_2 > 0$ 使

$$c_1\|x^{(k)} - x^*\|_\infty \leqslant \|x^{(k)} - x^*\| \leqslant c_2\|x^{(k)} - x^*\|_\infty,$$

于是又有

$$\lim_{k\to\infty}\|x^{(k)} - x^*\|_\infty = 0 \Leftrightarrow \lim_{k\to\infty}\|x^{(k)} - x^*\| = 0.$$

下面我们将向量范数概念推广到矩阵上去. 视 $\mathbf{R}^{n\times n}$ 中的矩阵为 $\mathbf{R}^{n^2}$ 中的向量，则由 $\mathbf{R}^{n^2}$ 上的 2-范数可以得到 $\mathbf{R}^{n\times n}$ 中矩阵的一种范数

$$F(\boldsymbol{A}) = \|\boldsymbol{A}\|_F = \left(\sum_{i,j=1}^{n} a_{i,j}^2\right)^{\frac{1}{2}},$$

称为 $\boldsymbol{A}$ 的 Frobenius 范数. $\|\boldsymbol{A}\|_F$ 显然满足正定性、齐次性及三角不等式.

下面我们给出矩阵范数的一般定义.

定义 4(矩阵的范数) 如果矩阵 $A\in\mathbf{R}^{n\times n}$ 的某个非负的实值函数 $N(A)=\|A\|$,满足条件

(1) $\|A\|\geqslant 0(\|A\|=0\Leftrightarrow A=0)$ (正定条件);

(2) $\|cA\|=|c|\,\|A\|$,c 为实数(齐次条件); (5.4)

(3) $\|A+B\|\leqslant\|A\|+\|B\|$ (三角不等式);

(4) $\|AB\|\leqslant\|A\|\,\|B\|$.

则称 $N(A)$ 是 $\mathbf{R}^{n\times n}$ 上的一个矩阵范数(或模).

上面我们定义的 $F(A)=\|A\|_F$ 就是 $\mathbf{R}^{n\times n}$ 上的一个矩阵范数.

由于在大多数与估计有关的问题中,矩阵和向量会同时参与讨论,所以希望引进一种矩阵的范数,它是和向量范数相联系而且和向量范数相容的,即对任何向量 $x\in\mathbf{R}^n$ 及 $A\in\mathbf{R}^{n\times n}$ 都成立

$$\|Ax\|\leqslant\|A\|\,\|x\|. \tag{5.5}$$

为此我们再引进一种矩阵的范数.

定义 5(矩阵的算子范数) 设 $x\in\mathbf{R}^n$, $A\in\mathbf{R}^{n\times n}$,给出一种向量范数 $\|x\|_v$(如 $v=1,2$ 或∞),相应地定义一个矩阵的非负函数

$$\|A\|_v=\max_{x\neq 0}\frac{\|Ax\|_v}{\|x\|_v}. \tag{5.6}$$

可验证 $\|A\|_v$ 满足定义 4(见下面定理),所以 $\|A\|_v$ 是 $\mathbf{R}^{n\times n}$ 上矩阵的一个范数,称为 A 的算子范数.

定理 17 设 $\|x\|_v$ 是 $\mathbf{R}^n$ 上一个向量范数,则 $\|A\|_v$ 是 $\mathbf{R}^{n\times n}$ 上矩阵的范数,且满足相容条件

$$\|Ax\|_v\leqslant\|A\|_v\|x\|_v. \tag{5.7}$$

证明 由(5.6)相容性条件(5.7)是显然的. 现只验证定义 4 中条件(4).

由(5.7),有

$$\|ABx\|_v\leqslant\|A\|_v\|Bx\|_v\leqslant\|A\|_v\|B\|_v\|x\|_v.$$

当 $\boldsymbol{x}\neq\boldsymbol{0}$ 时，有

$$\frac{\|\boldsymbol{ABx}\|_v}{\|\boldsymbol{x}\|_v}\leqslant\|\boldsymbol{A}\|_v\|\boldsymbol{B}\|_v,$$

故

$$\|\boldsymbol{AB}\|_v=\max_{x\neq 0}\frac{\|\boldsymbol{ABx}\|_v}{\|\boldsymbol{x}\|_v}\leqslant\|\boldsymbol{A}\|_v\|\boldsymbol{B}\|_v.$$

显然这种矩阵的范数 $\|\boldsymbol{A}\|_v$ 依赖于向量范数 $\|\boldsymbol{x}\|_v$ 的具体含义. 也就是说，当给出一种具体的向量范数 $\|\boldsymbol{x}\|_v$ 时，相应地就得到了一种矩阵范数 $\|\boldsymbol{A}\|_v$.

定理 18　设 $\boldsymbol{x}\in\mathbf{R}^n$，$\boldsymbol{A}\in\mathbf{R}^{n\times n}$，则

1. $\|\boldsymbol{A}\|_\infty=\max\limits_{1\leqslant i\leqslant n}\sum\limits_{j=1}^{n}|a_{ij}|$　（称为 $\boldsymbol{A}$ 的行范数），

2. $\|\boldsymbol{A}\|_1=\max\limits_{1\leqslant j\leqslant n}\sum\limits_{i=1}^{n}|a_{ij}|$　（称为 $\boldsymbol{A}$ 的列范数），

3. $\|\boldsymbol{A}\|_2=\sqrt{\lambda_{\max}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})}$　（称为 $\boldsymbol{A}$ 的 2- 范数），

其中 $\lambda_{\max}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})$ 表示 $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$ 的最大特征值.

证明　只就 1,3 给出证明，2 同理.

1. 设 $\boldsymbol{x}=(x_1,\cdots,x_n)^{\mathrm{T}}\neq\boldsymbol{0}$，不妨设 $\boldsymbol{A}\neq\boldsymbol{0}$. 记

$$t=\|\boldsymbol{x}\|_\infty=\max_{1\leqslant i\leqslant n}|x_i|,\ \mu=\max_{1\leqslant i\leqslant n}\sum_{j=1}^{n}|a_{ij}|,$$

则

$$\|\boldsymbol{Ax}\|_\infty=\max_{1\leqslant i\leqslant n}\left|\sum_{j=1}^{n}a_{ij}x_j\right|\leqslant\max_i\sum_{j=1}^{n}|a_{ij}||x_j|$$

$$\leqslant t\max_i\sum_{j=1}^{n}|a_{ij}|.$$

这说明对任何非零 $\boldsymbol{x}\in\mathbf{R}^n$，有

$$\frac{\|\boldsymbol{Ax}\|_\infty}{\|\boldsymbol{x}\|_\infty}\leqslant\mu.\tag{5.8}$$

下面来说明有一向量 $\boldsymbol{x}_0 \neq \boldsymbol{0}$，使 $\dfrac{\|\boldsymbol{A}\boldsymbol{x}_0\|_\infty}{\|\boldsymbol{x}_0\|_\infty} = \mu$. 设 $\mu = \sum_{j=1}^{n} |a_{i_0 j}|$，取向量 $\boldsymbol{x}_0 = (x_1, \cdots, x_n)^{\mathrm{T}}$，其中 $x_j = \operatorname{sgn}(a_{i_0 j})$ $(j = 1, 2, \cdots, n)$. 显然 $\|\boldsymbol{x}_0\|_\infty = 1$，且 $\boldsymbol{A}\boldsymbol{x}_0$ 的第 i_0 个分量为 $\sum_{i=1}^{n} a_{i_0 j} x_j = \sum_{j=1}^{n} |a_{i_0 j}|$，这说明

$$\|\boldsymbol{A}\boldsymbol{x}_0\|_\infty = \max_{1 \leqslant i \leqslant n} \left| \sum_{j=1}^{n} a_{ij} x_j \right| = \sum_{j=1}^{n} |a_{i_0 j}| = \mu.$$

3. 由于对一切 $\boldsymbol{x} \in \mathbf{R}^n$，$\|\boldsymbol{A}\boldsymbol{x}\|_2^2 = (\boldsymbol{A}\boldsymbol{x}, \boldsymbol{A}\boldsymbol{x}) = (\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}, \boldsymbol{x}) \geqslant 0$，从而 $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$ 的特征值为非负实数，设为

$$\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_n \geqslant 0. \tag{5.9}$$

$\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$ 为对称矩阵，设 $\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_n$ 为 $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$ 的相应于(5.9)的特征向量且 $(\boldsymbol{u}_i, \boldsymbol{u}_j) = \delta_{ij}$，又设 $\boldsymbol{x} \in \mathbf{R}^n$ 为任一非零向量，于是有

$$\boldsymbol{x} = \sum_{i=1}^{n} c_i \boldsymbol{u}_i,$$

其中 c_i 为组合系数，则

$$\frac{\|\boldsymbol{A}\boldsymbol{x}\|_2^2}{\|\boldsymbol{x}\|_2^2} = \frac{(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{x}, \boldsymbol{x})}{(\boldsymbol{x}, \boldsymbol{x})} = \frac{\sum_{i=1}^{n} c_i^2 \lambda_1}{\sum_{i=1}^{n} c_i^2} \leqslant \lambda_1.$$

另一方面，取 $\boldsymbol{x} = \boldsymbol{u}_1$，则上式等号成立，故

$$\|\boldsymbol{A}\|_2 = \max_{\boldsymbol{x} \neq \boldsymbol{0}} \frac{\|\boldsymbol{A}\boldsymbol{x}\|_2}{\|\boldsymbol{x}\|_2} = \sqrt{\lambda_1} = \sqrt{\lambda_{\max}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})}.$$

由定理 18 看出，计算一个矩阵的 $\|\boldsymbol{A}\|_\infty$，$\|\boldsymbol{A}\|_1$ 还是比较容易的，而矩阵的 2-范数 $\|\boldsymbol{A}\|_2$ 在计算上不方便，但是矩阵的 2-范数具有许多好的性质，它在理论上是非常有用的.

例 7 设 $\boldsymbol{A} = \begin{pmatrix} 1 & -2 \\ -3 & 4 \end{pmatrix}$，计算 $\boldsymbol{A}$ 的各种范数.

解　$\|A\|_1 = 6$，$\|A\|_\infty = 7$，$\|A\|_F \approx 5.477$，

$\|A\|_2 = \sqrt{15+\sqrt{221}} \approx 5.46$.

我们指出，对于复矩阵(即 $A\in C^{n\times n}$)定理 18 中 1,2. 显然也成立，对于 3 应改为

$$\|A\|_2 = \max_{x\neq 0}\left(\frac{x^H A^H A x}{x^H x}\right)^{1/2} = \sqrt{\lambda_{\max}(A^H A)}.$$

定义 6　设 $A\in R^{n\times n}$ 的特征值为 $\lambda_i(i=1,2,\cdots,n)$，称

$$\rho(A) = \max_{1\leq i\leq n} |\lambda_i|$$

为 A 的**谱半径**.

定理 19(特征值上界)　设 $A\in R^{n\times n}$，则 $\rho(A)\leq \|A\|$，即 A 的谱半径不超过 A 的任何一种算子范数(对 $\|A\|_F$ 亦对).

证明　设 λ 是 A 的任一特征值，x 为相应的特征向量，则 $Ax=\lambda x$，由(5.7)得

$$|\lambda|\ \|x\| = \|\lambda x\| = \|Ax\| \leq \|A\|\ \|x\|,$$

注意到 $\|x\|\neq 0$，即得

$$|\lambda| \leq \|A\|.$$

定理 20　如果 $A\in R^{n\times n}$ 为对称矩阵，则 $\|A\|_2=\rho(A)$.

证明留作习题.

定理 21　如果 $\|B\|<1$，则 $I\pm B$ 为非奇异矩阵，且

$$\|(I\pm B)^{-1}\| \leq \frac{1}{1-\|B\|},$$

其中 $\|\cdot\|$ 是指矩阵的算子范数.

证明　用反证法. 若 $\det(I-B)=0$，则 $(I-B)x=0$ 有非零解，即存在 $x_0\neq 0$ 使 $Bx_0=x_0$，$\dfrac{\|Bx_0\|}{\|x_0\|}=1$，故 $\|B\|\geq 1$，与假设矛盾. 又由 $(I-B)(I-B)^{-1}=I$，有

$$(I-B)^{-1} = I + B(I-B)^{-1},$$

从而

$$\|(I\pm B)^{-1}\| \leqslant \|I\| + \|B\|\,\|(I\pm B)^{-1}\|,$$

$$\|(I-B)^{-1}\| \leqslant \frac{1}{1-\|B\|}.$$

5.6 误差分析

5.6.1 矩阵的条件数

考虑线性方程组

$$Ax = b,$$

其中设 A 为非奇异矩阵, x 为方程组的精确解.

由于 A(或 b)元素是测量得到的,或者是计算的结果,在第一种情况 A(或 b)常带有某些观测误差,在后一种情况 A(或 b)又包含有舍入误差. 因此我们处理的实际矩阵是 $A+\delta A$(或 $b+\delta b$),下面我们来研究数据 A(或 b)的微小误差对解的影响. 即考虑估计 $x-y$,其中 y 是 $(A+\delta A)y=b$ 的解.

首先考察一个例子.

例 8 设有方程组

$$\begin{pmatrix}1 & 1\\ 1 & 1.0001\end{pmatrix}\begin{pmatrix}x_1\\ x_2\end{pmatrix} = \begin{pmatrix}2\\ 2\end{pmatrix}. \tag{6.1}$$

记为 $Ax=b$, 它的精确解为 $x=(2,0)^{\mathrm{T}}$.

现在考虑常数项的微小变化对方程组解的影响,即考察方程组

$$\begin{pmatrix}1 & 1\\ 1 & 1.0001\end{pmatrix}\begin{pmatrix}y_1\\ y_2\end{pmatrix} = \begin{pmatrix}2\\ 2.0001\end{pmatrix}, \tag{6.2}$$

也可表示为 $A(x+\delta x)=b+\delta b$, 其中 $\delta b=(0,0.0001)^{\mathrm{T}}$, $y=x+\delta x$, x 为(6.1)的解. 显然方程组(6.2)的解为 $x+\delta x=(1,1)^{\mathrm{T}}$.

我们看到(6.1)的常数项 $\boldsymbol{b}$ 的第 2 个分量只有 $\frac{1}{10000}$ 的微小变化，方程组的解却变化很大. 这样的方程组称为病态方程组.

定义 7　如果矩阵 $\boldsymbol{A}$ 或常数项 $\boldsymbol{b}$ 的微小变化，引起方程组 $\boldsymbol{Ax}=\boldsymbol{b}$ 解的巨大变化，则称此方程组为“**病态**”**方程组**，矩阵 $\boldsymbol{A}$ 称为“**病态**”**矩阵**（相对于方程组而言），否则称方程组为“**良态**”**方程组**，$\boldsymbol{A}$ 称为“**良态**”**矩阵**.

应该注意，矩阵的“病态”性质是矩阵本身的特性，下面我们希望找出刻画矩阵“病态”性质的量. 设有方程组

$$\boldsymbol{Ax}=\boldsymbol{b}, \tag{6.3}$$

其中 $\boldsymbol{A}$ 为非奇异阵，$\boldsymbol{x}$ 为(6.3)的准确解. 以下我们研究方程组的系数矩阵 $\boldsymbol{A}$（或 $\boldsymbol{b}$）的微小误差（扰动）时对解的影响.

现设 $\boldsymbol{A}$ 是精确的，$\boldsymbol{b}$ 有误差 $\delta\boldsymbol{b}$，解为 $\boldsymbol{x}+\delta\boldsymbol{x}$，则

$$\boldsymbol{A}(\boldsymbol{x}+\delta\boldsymbol{x})=\boldsymbol{b}+\delta\boldsymbol{b},\ \delta\boldsymbol{x}=\boldsymbol{A}^{-1}\delta\boldsymbol{b},$$

$$\|\delta\boldsymbol{x}\|\leqslant\|\boldsymbol{A}^{-1}\|\ \|\delta\boldsymbol{b}\|. \tag{6.4}$$

由(6.3)

$$\|\boldsymbol{b}\|\leqslant\|\boldsymbol{A}\|\ \|\boldsymbol{x}\|,$$

$$\frac{1}{\|\boldsymbol{x}\|}\leqslant\frac{\|\boldsymbol{A}\|}{\|\boldsymbol{b}\|}\ (\text{设 } \boldsymbol{b}\neq\boldsymbol{0}). \tag{6.5}$$

于是由(6.4)及(6.5)，得

定理 22　设 $\boldsymbol{A}$ 是非奇异阵，$\boldsymbol{Ax}=\boldsymbol{b}\neq\boldsymbol{0}$，且

$$\boldsymbol{A}(\boldsymbol{x}+\delta\boldsymbol{x})=\boldsymbol{b}+\delta\boldsymbol{b},$$

则

$$\frac{\|\delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|}\leqslant\|\boldsymbol{A}^{-1}\|\ \|\boldsymbol{A}\|\ \frac{\|\delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|}.$$

上式给出了解的相对误差的上界，常数项 $\boldsymbol{b}$ 的相对误差在解中可能放大 $\|\boldsymbol{A}^{-1}\|\ \|\boldsymbol{A}\|$ 倍.

现设 $\boldsymbol{b}$ 是精确的，$\boldsymbol{A}$ 有微小误差（扰动）$\delta\boldsymbol{A}$，解为 $\boldsymbol{x}+\delta\boldsymbol{x}$，则

$$(\boldsymbol{A}+\delta\boldsymbol{A})(\boldsymbol{x}+\delta\boldsymbol{x})=\boldsymbol{b},$$

$$(\boldsymbol{A}+\delta\boldsymbol{A})\delta\boldsymbol{x}=-(\delta\boldsymbol{A})\boldsymbol{x}. \tag{6.6}$$

如果 $\delta\boldsymbol{A}$ 不受限制的话，$\boldsymbol{A}+\delta\boldsymbol{A}$ 可能奇异，而

$$(\boldsymbol{A}+\delta\boldsymbol{A})=\boldsymbol{A}(\boldsymbol{I}+\boldsymbol{A}^{-1}\delta\boldsymbol{A}),$$

由定理 21 知，当 $\|\boldsymbol{A}^{-1}\delta\boldsymbol{A}\|<1$ 时，$(\boldsymbol{I}+\boldsymbol{A}^{-1}\delta\boldsymbol{A})^{-1}$ 存在. 由(6.6)式

$$\delta\boldsymbol{x}=-(\boldsymbol{I}+\boldsymbol{A}^{-1}\delta\boldsymbol{A})^{-1}\boldsymbol{A}^{-1}(\delta\boldsymbol{A})\boldsymbol{x},$$

因此
$$\|\delta\boldsymbol{x}\|\leqslant\frac{\|\boldsymbol{A}^{-1}\|\,\|\delta\boldsymbol{A}\|\,\|\boldsymbol{x}\|}{1-\|\boldsymbol{A}^{-1}(\delta\boldsymbol{A})\|}.$$

设 $\|\boldsymbol{A}^{-1}\|\,\|\delta\boldsymbol{A}\|<1$，即得

$$\frac{\|\delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|}\leqslant\frac{\|\boldsymbol{A}^{-1}\|\,\|\boldsymbol{A}\|\,\dfrac{\|\delta\boldsymbol{A}\|}{\|\boldsymbol{A}\|}}{1-\|\boldsymbol{A}^{-1}\|\,\|\boldsymbol{A}\|\,\dfrac{\|\delta\boldsymbol{A}\|}{\|\boldsymbol{A}\|}}. \tag{6.7}$$

定理 23 设 $\boldsymbol{A}$ 为非奇异矩阵，$\boldsymbol{Ax}=\boldsymbol{b}\neq\boldsymbol{0}$，且

$$(\boldsymbol{A}+\delta\boldsymbol{A})(\boldsymbol{x}+\delta\boldsymbol{x})=\boldsymbol{b}.$$

如果 $\|\boldsymbol{A}^{-1}\|\,\|\delta\boldsymbol{A}\|<1$，则(6.7)式成立.

如果 $\delta\boldsymbol{A}$ 充分小，且在条件 $\|\boldsymbol{A}^{-1}\|\,\|\delta\boldsymbol{A}\|<1$ 下，那么(6.7)式说明矩阵 $\boldsymbol{A}$ 的相对误差 $\dfrac{\|\delta\boldsymbol{A}\|}{\|\boldsymbol{A}\|}$ 在解中可能放大 $\|\boldsymbol{A}^{-1}\|\,\|\boldsymbol{A}\|$ 倍.

总之，量 $\|\boldsymbol{A}^{-1}\|\,\|\boldsymbol{A}\|$ 愈小，由 $\boldsymbol{A}$(或 $\boldsymbol{b}$)的相对误差引起的解的相对误差就愈小；量 $\|\boldsymbol{A}^{-1}\|\,\|\boldsymbol{A}\|$ 愈大，解的相对误差就可能愈大. 所以量 $\|\boldsymbol{A}^{-1}\|\,\|\boldsymbol{A}\|$ 实际上刻画了解对原始数据变化的灵敏程度，即刻画了方程组的“病态”程度，于是引进下述定义：

定义 8 设 $\boldsymbol{A}$ 为非奇异阵，称数 $\mathrm{cond}(\boldsymbol{A})_v=\|\boldsymbol{A}^{-1}\|_v\|\boldsymbol{A}\|_v$ ($v=1,2$ 或∞)为矩阵 $\boldsymbol{A}$ 的**条件数**.

由此看出矩阵的条件数与范数有关.

矩阵的条件数是一个十分重要的概念，由上面讨论知，当 $\boldsymbol{A}$ 的条件数相对的大，即 $\mathrm{cond}(\boldsymbol{A})\gg1$ 时，则(6.3)是“病态”的(即 $\boldsymbol{A}$ 是

“病态”矩阵，或者说 $\boldsymbol{A}$ 是坏条件的，相对于解方程组)，当 $\boldsymbol{A}$ 的条件数相对的小，则(6.3)是“良态”的(或者说 $\boldsymbol{A}$ 是好条件的). 注意，方程组病态性质是方程组本身的特性. $\boldsymbol{A}$ 的条件数愈大，方程组的病态程度愈严重，也就愈难用一般的计算方法求得比较准确的解.

通常使用的条件数，有

(1) $\mathrm{cond}(\boldsymbol{A})_\infty = \|\boldsymbol{A}^{-1}\|_\infty \|\boldsymbol{A}\|_\infty$;

(2) $\boldsymbol{A}$ 的谱条件数

$$\mathrm{cond}(\boldsymbol{A})_2 = \|\boldsymbol{A}\|_2 \|\boldsymbol{A}^{-1}\|_2 = \sqrt{\frac{\lambda_{\max}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})}{\lambda_{\min}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})}}.$$

当 $\boldsymbol{A}$ 为对称矩阵时

$$\mathrm{cond}(\boldsymbol{A})_2 = \frac{|\lambda_1|}{|\lambda_n|},$$

其中 λ_1, λ_n 为 $\boldsymbol{A}$ 的绝对值最大和绝对值最小的特征值.

条件数的性质：

1. 对任何非奇异矩阵 $\boldsymbol{A}$，都有 $\mathrm{cond}(\boldsymbol{A})_v \geqslant 1$. 事实上，

$$\mathrm{cond}(\boldsymbol{A})_v = \|\boldsymbol{A}^{-1}\|_v \|\boldsymbol{A}\|_v \geqslant \|\boldsymbol{A}^{-1}\boldsymbol{A}\|_v = 1;$$

2. 设 $\boldsymbol{A}$ 为非奇异阵且 $c \neq 0$(常数)，则

$$\mathrm{cond}(c\boldsymbol{A})_v = \mathrm{cond}(\boldsymbol{A})_v;$$

3. 如果 $\boldsymbol{A}$ 为正交矩阵，则 $\mathrm{cond}(\boldsymbol{A})_2 = 1$；如果 $\boldsymbol{A}$ 为非奇异矩阵，$\boldsymbol{R}$ 为正交矩阵，则

$$\mathrm{cond}(\boldsymbol{RA})_2 = \mathrm{cond}(\boldsymbol{AR})_2 = \mathrm{cond}(\boldsymbol{A})_2.$$

例 9　已知希尔伯特(Hilbert)矩阵

$$\boldsymbol{H}_n = \begin{pmatrix} 1 & \frac{1}{2} & \cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n+1} \\ \vdots & \vdots & & \vdots \\ \frac{1}{n} & \frac{1}{1+n} & \cdots & \frac{1}{2n-1} \end{pmatrix},$$

计算 $\boldsymbol{H}_3$ 的条件数.

解

$$\boldsymbol{H}_3=\begin{pmatrix}1 & \frac{1}{2} & \frac{1}{3}\\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4}\\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5}\end{pmatrix},\quad \boldsymbol{H}_3^{-1}=\begin{pmatrix}9 & -36 & 30\\ -36 & 192 & -180\\ 30 & -180 & 180\end{pmatrix}.$$

(1) 计算 $\boldsymbol{H}_3$ 条件数 $\mathrm{cond}(\boldsymbol{H}_3)_\infty$

$\|\boldsymbol{H}_3\|_\infty=11/6$，$\|\boldsymbol{H}_3^{-1}\|_\infty=408$，所以 $\mathrm{cond}(\boldsymbol{H}_3)_\infty=748$. 同样可计算 $\mathrm{cond}(\boldsymbol{H}_6)_\infty=2.9\times10^7$，$\mathrm{cond}(\boldsymbol{H}_7)_\infty=9.85\times10^8$. 当 n 愈大时，$\boldsymbol{H}_n$ 矩阵病态愈严重.

(2) 考虑方程组

$$\boldsymbol{H}_3\boldsymbol{x}=(11/6,\ 13/12,\ 47/60)^{\mathrm{T}}=\boldsymbol{b},$$

设 $\boldsymbol{H}_3$ 及 $\boldsymbol{b}$ 有微小误差(取 3 位有效数字)有

$$\begin{pmatrix}1.00 & 0.500 & 0.333\\ 0.500 & 0.333 & 0.250\\ 0.333 & 0.250 & 0.200\end{pmatrix}\begin{pmatrix}x_1+\delta x_1\\ x_2+\delta x_2\\ x_3+\delta x_3\end{pmatrix}=\begin{pmatrix}1.83\\ 1.08\\ 0.783\end{pmatrix},\tag{6.8}$$

简记为 $(\boldsymbol{H}_3+\delta\boldsymbol{H}_3)(\boldsymbol{x}+\delta\boldsymbol{x})=\boldsymbol{b}+\delta\boldsymbol{b}$. 方程组 $\boldsymbol{H}_3\boldsymbol{x}=\boldsymbol{b}$ 与(6.8)的精确解分别为：$\boldsymbol{x}=(1,1,1)^{\mathrm{T}}$，$\boldsymbol{x}+\delta\boldsymbol{x}=(1.089512538,\ 0.487967062,\ 1.491002798)^{\mathrm{T}}$. 于是

$$\delta\boldsymbol{x}=(0.0895,\ -0.5120,\ 0.4910)^{\mathrm{T}},$$

$$\frac{\|\delta\boldsymbol{H}_3\|_\infty}{\|\boldsymbol{H}_3\|_\infty}\approx0.18\times10^{-3}<0.02\%,$$

$$\frac{\|\delta\boldsymbol{b}\|_\infty}{\|\boldsymbol{b}\|_\infty}\approx0.182\%,\quad\frac{\|\delta\boldsymbol{x}\|_\infty}{\|\boldsymbol{x}\|_\infty}\approx51.2\%.$$

这就是说 $\boldsymbol{H}_3$ 与 $\boldsymbol{b}$ 相对误差不超过 0.3%，而引起解的相对误差超过 50%.

由上面的讨论，要判别一个矩阵是否病态需要计算条件数 $\mathrm{cond}(\boldsymbol{A})=\|\boldsymbol{A}^{-1}\|\,\|\boldsymbol{A}\|$，而计算 $\boldsymbol{A}^{-1}$ 是比较费劲的，那么在实际计算中如何发现病态情况呢？

(1) 如果在 $\boldsymbol{A}$ 的三角约化时(尤其是用主元素消去法解(6.3)时)出现小主元，对大多数矩阵来说，$\boldsymbol{A}$ 是病态矩阵. 例如用选主元的直接三角分解法解方程组(6.8)(结果舍入为 3 位浮点数)，则有

$$\boldsymbol{I}_{23}(\boldsymbol{H}_3+\delta\boldsymbol{H}_3)=\begin{pmatrix}1 & & \\ 0.333 & 1 & \\ 0.500 & 0.994 & 1\end{pmatrix}\times\begin{pmatrix}1 & 0.5000 & 0.3330\\ & 0.0835 & 0.0891\\ & & -0.00507\end{pmatrix}=\boldsymbol{LU}.$$

(2) 系数矩阵的行列式值相对说很小，或系数矩阵某些行近似线性相关，这时 $\boldsymbol{A}$ 可能病态.

(3) 系数矩阵 $\boldsymbol{A}$ 元素间数量级相差很大，并且无一定规则，$\boldsymbol{A}$ 可能病态.

用选主元素的消去法不能解决病态问题，对于病态方程组可采用高精度的算术运算(采用双倍字长进行运算)或者采用预处理方法. 即将求解 $\boldsymbol{Ax}=\boldsymbol{b}$ 转化为一等价方程组

$$\begin{cases}\boldsymbol{PAQy}=\boldsymbol{Pb};\\ \boldsymbol{y}=\boldsymbol{Q}^{-1}\boldsymbol{x}.\end{cases}$$

选择非奇异矩阵 $\boldsymbol{P},\boldsymbol{Q}$ 使

$$\mathrm{cond}(\boldsymbol{PAQ})<\mathrm{cond}(\boldsymbol{A}).$$

一般选择 $\boldsymbol{P},\boldsymbol{Q}$ 为对角阵或者三角矩阵.

当矩阵 $\boldsymbol{A}$ 的元素大小不均时，对 $\boldsymbol{A}$ 的行(或列)引进适当的比例因子(使矩阵 $\boldsymbol{A}$ 的所有行或列按∞-范数大体上有相同的长度，

使 $\boldsymbol{A}$ 的系数均衡),对 $\boldsymbol{A}$ 的条件数是有影响的. 这种方法不能保证 $\boldsymbol{A}$ 的条件数一定得到改善.

例 10 设

$$\begin{pmatrix} 1 & 10^4 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 10^4 \\ 2 \end{pmatrix}, \tag{6.9}$$

计算 $\mathrm{cond}(\boldsymbol{A})_\infty$.

$$\boldsymbol{A} = \begin{pmatrix} 1 & 10^4 \\ 1 & 1 \end{pmatrix},\ \boldsymbol{A}^{-1} = \frac{1}{10^4 - 1}\begin{pmatrix} -1 & 10^4 \\ 1 & -1 \end{pmatrix},$$

$$\mathrm{cond}(\boldsymbol{A})_\infty = \frac{(1+10^4)^2}{10^4 - 1} \approx 10^4.$$

现在 $\boldsymbol{A}$ 的第一行引进比例因子. 如用 $s_1 = \max\limits_{1 \leqslant i \leqslant 2} |a_{1i}| = 10^4$ 除第一个方程式,得 $\boldsymbol{A}'\boldsymbol{x} = \boldsymbol{b}'$,即

$$\begin{pmatrix} 10^{-4} & 1 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \tag{6.10}$$

而
$$(\boldsymbol{A}')^{-1} = \frac{1}{1 - 10^{-4}}\begin{pmatrix} -1 & 1 \\ 1 & -10^{-4} \end{pmatrix},$$

于是

$$\mathrm{cond}(\boldsymbol{A}')_\infty = \frac{1}{1 - 10^{-4}} \approx 4.$$

当用列主元消去法解(6.9)时(计算到三位数字),

$$(\boldsymbol{A} \mid \boldsymbol{b}) \to \left(\begin{array}{cc:c} 1 & 10^4 & 10^4 \\ 0 & -10^4 & -10^4 \end{array}\right),$$

于是得到很坏的结果: $x_2 = 1$, $x_1 = 0$.

现用列主元消去法解(6.10),得到

$$(\boldsymbol{A}' \mid \boldsymbol{b}') \to \left(\begin{array}{cc:c} 1 & 1 & 2 \\ 10^{-4} & 1 & 1 \end{array}\right) \to \left(\begin{array}{cc:c} 1 & 1 & 2 \\ 0 & 1 & 1 \end{array}\right),$$

从而得到较好的计算解: $x_1 = 1$, $x_2 = 1$.

设 $\bar{x}$ 为方程组 $Ax=b$ 的近似解，于是可计算 $\bar{x}$ 的剩余向量 $r=b-A\bar{x}$，当 r 很小时，$\bar{x}$ 是否为 $Ax=b$ 一个较好的近似解呢？下述定理给出了解答.

定理 24（事后误差估计） 设 A 为非奇异矩阵，x 是 $Ax=b\neq 0$ 的精确解. 再设 $\bar{x}$ 是此方程组的近似解，$r=b-A\bar{x}$，则

$$\frac{\|x-\bar{x}\|}{\|x\|}\leqslant \mathrm{cond}(A)\cdot\frac{\|r\|}{\|b\|}. \tag{6.11}$$

证明 由 $x-\bar{x}=A^{-1}r$，得

$$\|x-\bar{x}\|\leqslant\|A^{-1}\|\,\|r\|, \tag{6.12}$$

又有

$$\|b\|=\|Ax\|\leqslant\|A\|\,\|x\|,\quad \frac{1}{\|x\|}\leqslant\frac{\|A\|}{\|b\|}, \tag{6.13}$$

由(6.12)及(6.13)即得到(6.11).

(6.11)式说明，近似解 $\bar{x}$ 的精度（误差界）不仅依赖于剩余 r 的"大小"，而且依赖于 A 的条件数. 当 A 是病态时，即使有很小的剩余 r，也不能保证 $\bar{x}$ 是高精度的近似解.

5.6.2 迭代改善法

设 $Ax=b$，其中 $A\in\mathbf{R}^{n\times n}$ 为非奇异矩阵，且为病态方程组（但不过分病态）. 当求得方程组的近似解 x_1，下面研究改善方程组近似解 x_1 精度的方法.

首先用选主元三角分解法实现分解计算

$$PA\doteq LU,$$

其中 P 为置换阵，L 为单位下三角阵，U 为上三角阵，且求得计算解 x_1.

现利用 x_1 的剩余向量来提高 x_1 的精度.

计算剩余向量

$$r_1 = b - Ax_1, \tag{6.14}$$

求解 $Ad=r_1$，得到的解记为 d_1. 然后改善

$$x_2 = x_1 + d_1. \tag{6.15}$$

显然，如果(6.14)，(6.15)及解 $Ad=r_1$ 的计算没有误差，则 x_2 就是 $Ax=b$ 的精确解. 事实上，

$$Ax_2 = A(x_1 + d_1) = Ax_1 + Ad_1 = Ax_1 + r_1 = b.$$

但是，在实际计算中，由于有舍入误差，x_2 只是方程组的近似解，重复(6.14)，(6.15)过程，就产生一近似解序列 $\{x_k\}$，有时可能得到比较好的近似.

算法 5(迭代改善法) 设 $Ax=b$，其中 $A\in \mathbf{R}^{n\times n}$ 为非奇异矩阵，且 $Ax=b$ 为病态方程组(但不过分病态)，用选主元分解法实现 $PA\doteq LU$ 及计算解 x_1. 本算法用迭代改善法提高近似解 x_1 精度. 设计算机字长为 t，用数组 $A(n,n)$ 保存 A 元素，数组 $C(n,n)$ 保存三角矩阵 L 及 U，用 $\mathrm{Ip}(n)$ 记录行交换信息，$x(n)$ 存贮 x_1 及 x_k，$r(n)$ 保存 r_k 或 d_k.

1. 用选主元三角分解实行分解计算

$PA\doteq LU$ 且求计算解 x_1(用单精度)

2. 对于 $k=1,2,\cdots,N_0$

(1) 计算 $r_k=b-Ax_k$(用原始 A 及双精度计算)

(2) 求解 $LUd_k=Pr_k$，即 $\begin{cases} Ly=Pr_k, \\ Ud_k=y. \end{cases}$(用单精度计算)

(3) 如果 $\|d_k\|_\infty/\|x_k\|_\infty\leqslant 10^{-t}$ 则输出 k,x_k,r_k，停机

(4) 改善 $x_{k+1}=x_k+d_k$ (用单精度计算)

3. 输出迭代改善方法迭代 N_0 次失败信息

当 $Ax=b$ 不是过分病态时，迭代改善法是比较好的改进近似解精度的一种方法，当 $Ax=b$ 非常病态时，$\{x_k\}$ 可能不收敛.

迭代改善法的实现要依赖于机器及需要保留 $\boldsymbol{A}$ 的原始副本.

例 11 用迭代改善法解

$$\begin{bmatrix} 1.0303 & 0.99030 \\ 0.99030 & 0.95285 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2.4944 \\ 2.3988 \end{bmatrix} \quad (\text{记为 } \boldsymbol{Ax} = \boldsymbol{b})$$

(这里 $\beta=10$, $t=5$,用5位浮点数运算).

解 精确解 $\boldsymbol{x}^* = (1.2240, 1.2454)^{\mathrm{T}}$(舍入到小数后第4位).

容易计算

$$\mathrm{cond}(\boldsymbol{A})_\infty = \|\boldsymbol{A}\|_\infty \|\boldsymbol{A}^{-1}\|_\infty \doteq 2 \times 2000 = 4000.$$

首先实现分解计算 $\boldsymbol{A} \doteq \boldsymbol{LU}$,且求 $\boldsymbol{x}_1$.

$$\boldsymbol{A} \doteq \begin{bmatrix} 1 & 0 \\ 0.9118 & 1 \end{bmatrix}\begin{bmatrix} 1.0303 & 0.99030 \\ 0 & 0.00099 \end{bmatrix} = \boldsymbol{LU},$$

且得计算解 $\boldsymbol{x}_1 = (1.2560, 1.2121)^{\mathrm{T}}$.

应用迭代改善法需要用原始矩阵 $\boldsymbol{A}$ 且用双倍字长精度计算剩余向量 $\boldsymbol{r} = \boldsymbol{b} - \boldsymbol{Ax}$,其他计算用单精度. 计算如下表.

$\boldsymbol{x}_1$	$\boldsymbol{r}_1$	$\boldsymbol{d}_1$	$\boldsymbol{x}_2$	$\boldsymbol{r}_2$	$\boldsymbol{d}_2$
1.2560	5.7×10^{-7}	-0.03220	1.2238	1.18×10^{-6}	2.285×10^{-4}
1.2121	3.3715×10^{-5}	0.033502	1.2456	9×10^{-7}	-2.365×10^{-4}

$$\boldsymbol{x}_3 = (1.2240, 1.2454)^{\mathrm{T}},$$

$$\boldsymbol{r}_3 = (-0.682\times10^{-5}, -0.659\times10^{-5})^{\mathrm{T}},$$

$$\boldsymbol{d}_3 = (0.2717\times10^{-4}, -0.3515\times10^{-4})^{\mathrm{T}}.$$

如果 $\boldsymbol{x}_k$ 需要更多的数位,迭代可以继续.

5.7 矩阵的正交三角化及应用

本节介绍初等反射阵及平面旋转阵,矩阵正交约化,它们在矩

阵计算中起着重要作用.

5.7.1 初等反射阵

定义 9 设向量 $w\in\mathbf{R}^n$ 且 $w^{\mathrm{T}}w=1$,称矩阵

$$H(w)=I-2ww^{\mathrm{T}}$$

为初等反射阵(或称为豪斯霍尔德(Householder)变换). 如果记 $w=(w_1,w_2,\cdots,w_n)^{\mathrm{T}}$, 则

$$H(w)=\begin{pmatrix}1-2w_1^2 & -2w_1w_2 & \cdots & -2w_1w_n\\ -2w_2w_1 & 1-2w_2^2 & \cdots & -2w_2w_n\\ \vdots & \vdots & & \vdots\\ -2w_nw_1 & -2w_nw_2 & \cdots & 1-2w_n^2\end{pmatrix}.$$

定理 25 设有初等反射阵 $H=I-2ww^{\mathrm{T}}$, 其中 $w^{\mathrm{T}}w=1$,则:

(1) H 是对称矩阵,即 $H^{\mathrm{T}}=H$.

(2) H 是正交矩阵,即 $H^{-1}=H$.

(3) 设 A 为对称矩阵,那么 $A_1=H^{-1}AH=HAH$ 亦是对称矩阵.

证明 只证 H 的正交性,其他显然.

$$\begin{aligned}H^{\mathrm{T}}H&=H^2=(I-2ww^{\mathrm{T}})(I-2ww^{\mathrm{T}})\\&=I-4ww^{\mathrm{T}}+4w(w^{\mathrm{T}}w)w^{\mathrm{T}}=I.\end{aligned}$$

设向量 $u\neq\mathbf{0}$, 则显然

$$H=I-2\frac{uu^{\mathrm{T}}}{\|u\|_2^2}$$

是一个初等反射阵.

下面考察初等反射阵的几何意义. 考虑以 w 为法向量且过原点 O 的超平面 S: $w^{\mathrm{T}}x=0$. 设任意向量 $v\in\mathbf{R}^n$,则 $v=x+y$,其中 $x\in S$,$y\in S^{\perp}$. 于是

$$Hx=(I-2ww^{\mathrm{T}})x=x-2ww^{\mathrm{T}}x=x.$$

对于 $y\in S^{\perp}$，易知 $Hy=-y$，从而对任意向量 $v\in \mathbf{R}^n$，总有

$$Hv = x - y = v',$$

其中 v' 为 v 关于平面 S 的镜面反射(见图 5-1).

初等反射矩阵在计算上的意义是它能用来约化矩阵，例如设向量 $x\neq \mathbf{0}$，可选择一初等反射阵 H 使 $Hx=\sigma e_1$. 为此给出下面定理.

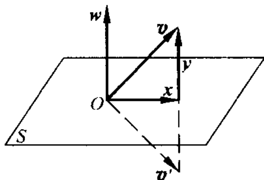


图　5-1

定理 26　设 x,y 为两个不相等的 n 维向量，$\|x\|_2=\|y\|_2$，则存在一个初等反射阵 H，使 $Hx=y$.

证明　令 $w=\dfrac{x-y}{\|x-y\|_2}$，则得到一个初等反射阵

$$H = I - 2ww^{\mathrm{T}} = I - 2\frac{(x-y)}{\|x-y\|_2^2}(x^{\mathrm{T}} - y^{\mathrm{T}}),$$

而且

$$\begin{aligned} Hx &= x - 2\frac{x-y}{\|x-y\|_2^2}(x^{\mathrm{T}} - y^{\mathrm{T}})x \\ &= x - 2\frac{(x-y)(x^{\mathrm{T}}x - y^{\mathrm{T}}x)}{\|x-y\|_2^2}. \end{aligned}$$

因为

$$\|x-y\|_2^2 = (x-y)^{\mathrm{T}}(x-y) = 2(x^{\mathrm{T}}x - y^{\mathrm{T}}x),$$

所以

$$Hx = x - (x-y) = y.$$

容易说明，w 是使 $Hx=y$ 成立的唯一长度等于 1 的向量(不计符号).

定理 27(约化定理)　设 $x=(x_1,x_2,\cdots,x_n)^{\mathrm{T}}\neq \mathbf{0}$，则存在初等反射阵 H 使 $Hx=-\sigma e_1$，其中

$$\begin{cases} \boldsymbol{H} = \boldsymbol{I} - \beta^{-1}\boldsymbol{u}\boldsymbol{u}^{\mathrm{T}}, \\ \sigma = \operatorname{sgn}(x_1)\,\|\boldsymbol{x}\|_2, \\ \boldsymbol{u} = \boldsymbol{x} + \sigma\boldsymbol{e}_1, \\ \beta = \dfrac{1}{2}\|\boldsymbol{u}\|_2^2 = \sigma(\sigma + x_1). \end{cases}$$

证明 记 $\boldsymbol{y}=-\sigma\boldsymbol{e}_1$，设 $\boldsymbol{x}\neq\boldsymbol{y}$，取 $\sigma=\pm\|\boldsymbol{x}\|_2$，则有 $\|\boldsymbol{x}\|_2=\|\boldsymbol{y}\|_2$，于是由定理 26 存在 $\boldsymbol{H}$ 变换

$$\boldsymbol{H} = \boldsymbol{I} - 2\boldsymbol{w}\boldsymbol{w}^{\mathrm{T}},$$

其中 $\boldsymbol{w}=\dfrac{\boldsymbol{x}+\sigma\boldsymbol{e}_1}{\|\boldsymbol{x}+\sigma\boldsymbol{e}_1\|_2}$，使 $\boldsymbol{H}\boldsymbol{x}=\boldsymbol{y}=-\sigma\boldsymbol{e}_1$.

记 $\boldsymbol{u}=\boldsymbol{x}+\sigma\boldsymbol{e}_1\equiv(u_1,u_2,\cdots,u_n)^{\mathrm{T}}$. 于是

$$\boldsymbol{H} = \boldsymbol{I} - 2\,\frac{\boldsymbol{u}\boldsymbol{u}^{\mathrm{T}}}{\|\boldsymbol{u}\|_2^2} = \boldsymbol{I} - \beta^{-1}\boldsymbol{u}\boldsymbol{u}^{\mathrm{T}},$$

其中 $\boldsymbol{u}=(x_1+\sigma,x_2,\cdots,x_n)^{\mathrm{T}}$，$\beta=\dfrac{1}{2}\|\boldsymbol{u}\|_2^2$.

显然

$$\begin{aligned} \beta &= \frac{1}{2}\|\boldsymbol{u}\|_2^2 = \frac{1}{2}((x_1+\sigma)^2 + x_2^2 + \cdots + x_n^2) \\ &= \sigma(\sigma + x_1). \end{aligned}$$

如果 σ 和 x_1 异号，那么计算 $x_1+\sigma$ 时有效数字可能损失，我们取 σ 和 x_1 有相同的符号，即取

$$\sigma = \operatorname{sgn}(x_1)\,\|\boldsymbol{x}\|_2 = \operatorname{sgn}(x_1)\Big(\sum_{i=1}^{n} x_i^2\Big)^{1/2}.$$

在计算 σ 时，可能上溢或下溢，为了避免溢出，将 $\boldsymbol{x}$ 规范化

$$d = \|\boldsymbol{x}\|_\infty,\ \boldsymbol{x}' = \frac{\boldsymbol{x}}{d} \quad (\text{设 } d \neq 0),$$

则有 $\boldsymbol{H}'$ 使 $\boldsymbol{H}'\boldsymbol{x}'=\sigma'\boldsymbol{e}_1$，其中

$$\begin{cases} \boldsymbol{H}' = \boldsymbol{I} - (\beta')^{-1}\boldsymbol{u}'\boldsymbol{u}'^{\mathrm{T}}, \\ \sigma' = \sigma/d,\ \boldsymbol{u}' = \boldsymbol{u}/d,\ \beta' = \beta/d^2, \\ \boldsymbol{H}' = \boldsymbol{H}. \end{cases}$$

算法 6 设 $\boldsymbol{x}=(x_1,x_2,\cdots,x_n)^{\mathrm{T}}\neq\boldsymbol{0}$，本算法计算 $\boldsymbol{u}'$，β' 及 σ 使 $(\boldsymbol{I}-\beta'^{-1}\boldsymbol{u}'\boldsymbol{u}'^{\mathrm{T}})\boldsymbol{x}=-\sigma\boldsymbol{e}_1$，$\boldsymbol{u}'$ 的分量冲掉 $\boldsymbol{x}$ 的分量.

(1) 计算 d $\quad d=\max\limits_{1\leqslant i\leqslant n}|x_i|$

(2) $x_i\leftarrow u_i=x_i/d \quad (i=1,2,\cdots,n)$

(3) $\sigma\leftarrow\mathrm{sgn}(u_1)\sqrt{\sum\limits_{i=1}^{n}u_i^2}$

(4) $u_1\leftarrow u_1+\sigma$

(5) $\beta\leftarrow\sigma*u_1$

(6) $\sigma\leftarrow d*\sigma$

关于 $\boldsymbol{HA}$ 的计算，设 $\boldsymbol{A}=(\boldsymbol{a}_1\ \boldsymbol{a}_2\cdots\ \boldsymbol{a}_n)\in\mathbf{R}^{m\times n}$，其中 $\boldsymbol{a}_i$ 为 $\boldsymbol{A}$ 的第 i 列向量，则

$$\boldsymbol{HA}=(\boldsymbol{Ha}_1\quad \boldsymbol{Ha}_2\ \cdots\ \boldsymbol{Ha}_n),$$

因此计算 $\boldsymbol{HA}$ 就是要计算

$$\boldsymbol{Ha}_i=(\boldsymbol{I}-\beta^{-1}\boldsymbol{uu}^{\mathrm{T}})\boldsymbol{a}_i=\boldsymbol{a}_i-(\beta^{-1}\boldsymbol{u}^{\mathrm{T}}\boldsymbol{a}_i)\boldsymbol{u}\quad (i=1,2,\cdots,n).$$

于是计算 $\boldsymbol{Ha}_i$ 只需计算两向量的数量积和两向量的加法. 计算 $\boldsymbol{HA}$ 只需作 $2nm$ 次乘法运算.

5.7.2 平面旋转矩阵

设 $\boldsymbol{x},\boldsymbol{y}\in\mathbf{R}^2$，则变换

$$\begin{pmatrix}y_1\\y_2\end{pmatrix}=\begin{pmatrix}\cos\theta & \sin\theta\\-\sin\theta & \cos\theta\end{pmatrix}\begin{pmatrix}x_1\\x_2\end{pmatrix},\quad \text{或}\quad \boldsymbol{y}=\boldsymbol{Px}$$

是平面上向量的一个旋转变换，其中

$$\boldsymbol{P}(\theta)=\begin{pmatrix}\cos\theta & \sin\theta\\-\sin\theta & \cos\theta\end{pmatrix}$$

为正交矩阵.

$\mathbf{R}^n$ 中变换：$\boldsymbol{y}=\boldsymbol{Px}$，

其中 $\boldsymbol{x}=(x_1,x_2,\cdots,x_n)^{\mathrm{T}}$，$\boldsymbol{y}=(y_1,y_2,\cdots,y_n)^{\mathrm{T}}$，而

$$
\boldsymbol{P}\equiv\boldsymbol{P}(i,j,\theta)=\begin{array}{c}\begin{matrix}&&&i&&&j&&&\end{matrix}\\\begin{pmatrix}
1 &&&&&&&&&\\
&\ddots &&&&&&&&\\
&&1&&&&&&&\\
&&&\cos\theta&&&\sin\theta&&&\\
&&&&1&&&&&\\
&&&&&\ddots&&&&\\
&&&&&&1&&&\\
&&&-\sin\theta&&&\cos\theta&&&\\
&&&&&&&1&&\\
&&&&&&&&\ddots&\\
&&&&&&&&&1
\end{pmatrix}\end{array}\begin{matrix}\\ \\ \\ i\\ \\ \\ \\ j\\ \\ \\ \\ \end{matrix}
$$

称为 $\mathbf{R}^n$ 中平面 $\{x_i,x_j\}$ 的旋转变换（或称为吉文斯(Givens)变换），$\boldsymbol{P}=\boldsymbol{P}(i,j,\theta)=P(i,j)$ 称为平面旋转矩阵.

显然，$\boldsymbol{P}(i,j,\theta)$ 具有性质：

(1) $\boldsymbol{P}$ 与单位阵 $\boldsymbol{I}$ 只是在 (i,i)，(i,j)，(j,i)，(j,j) 位置元素不一样，其他相同.

(2) $\boldsymbol{P}$ 为正交矩阵($\boldsymbol{P}^{-1}=\boldsymbol{P}^{\mathrm{T}}$).

(3) $\boldsymbol{P}(i,j)\boldsymbol{A}$（左乘）只需计算第 i 行与第 j 行元素，即对 $\boldsymbol{A}=(a_{ij})_{m\times n}$ 有

$$
\begin{pmatrix}a'_{il}\\ a'_{jl}\end{pmatrix}=\begin{pmatrix}c & s\\ -s & c\end{pmatrix}\begin{pmatrix}a_{il}\\ a_{jl}\end{pmatrix}\quad(l=1,2,\cdots,n).
$$

其中 $c=\cos\theta$，$s=\sin\theta$.

(4) $\boldsymbol{AP}(i,j)$（右乘）只需计算第 i 列与第 j 列元素

$$
(a'_{li},\ a'_{lj})=(a_{li},\ a_{lj})\begin{pmatrix}c & s\\ -s & c\end{pmatrix}\quad(l=1,2,\cdots,m).
$$

利用平面旋转变换，可使向量 $\boldsymbol{x}$ 中的指定元素变为零.

定理 28（约化定理）　设 $x=(x_1,\cdots,x_i,\cdots,x_j,\cdots,x_n)^T$，其中 x_i，x_j 不全为零，则可选择平面旋转阵 $P(i,j,\theta)$，使

$$Px = (x_1,\cdots,\overset{i}{x_i'},\cdots,\overset{j}{0},\cdots,x_n)^T,$$

其中 $x_i'=\sqrt{x_i^2+x_j^2}$，$\theta=\arctan(x_j/x_i)$.

证明　取 $c=\cos\theta=x_i/x_i'$，$s=\sin\theta=x_j/x_i'$. 由 $P(i,j,\theta)x=x'=(x_1',x_2',\cdots,x_i',\cdots,x_j',\cdots,x_n')^T$，利用矩阵乘法，显然有

$$\begin{cases} x_i' = cx_i + sx_j, \\ x_j' = -sx_i + cx_j, \\ x_k' = x_k \quad (k \neq i,j). \end{cases}$$

于是，由 c,s 的取法得

$$x_i' = \sqrt{x_i^2 + x_j^2},\ x_j' = 0.$$

5.7.3　矩阵的 QR 分解

下面讨论用正交矩阵来约化矩阵，可得到下述结果.

设 $A\in \mathbf{R}^{m\times n}$ 且为非零矩阵，则存在初等反射阵 $H_1,H_2,\cdots,H_s$ 使

$$H_s\cdots H_2H_1A = A^{(s+1)}\quad（上梯形）.$$

设有

$$A = A^{(1)} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} = (a_1\ a_2\ \cdots\ a_n)\quad（按列分块）. \tag{7.1}$$

(1) 第 1 步约化：如果 $a_1=0$，取 $H_1=I$，即这一步不需要约化，不妨设 $a_1\neq 0$，于是可选取初等反射阵 $H_1=I-\beta_1^{-1}u_1u_1^T$ 使

$$\boldsymbol{H}_1\boldsymbol{a}_1 = -\sigma_1\boldsymbol{e}_1.$$

于是

$$\boldsymbol{H}_1\boldsymbol{A}^{(1)} = (\boldsymbol{H}_1\boldsymbol{a}_1 \quad \boldsymbol{H}_1\boldsymbol{a}_2 \quad \cdots \quad \boldsymbol{H}_1\boldsymbol{a}_n)$$

$$= \begin{pmatrix} -\sigma_1 & a_{12}^{(2)} & \cdots & a_{1n}^{(2)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{m2}^{(2)} & \cdots & a_{mn}^{(2)} \end{pmatrix} = \begin{pmatrix} -\sigma_1 & r_2 & \boldsymbol{B}_2 \\ \boldsymbol{0} & \boldsymbol{c}_2 & \boldsymbol{D}_2 \end{pmatrix},$$

其中 $\boldsymbol{c}_2 = (a_{22}^{(2)}, \cdots, a_{m2}^{(2)})^{\mathrm{T}} \in \mathbf{R}^{m-1}$，$\boldsymbol{D}_2 \in \mathbf{R}^{(m-1)\times(n-2)}$.

(2) 第 k 步约化：设已完成对 $\boldsymbol{A}$ 上述第 1 步…第 $k-1$ 步的约化，再进行第 k 步约化. 即存在初等反射阵 $\boldsymbol{H}_1, \boldsymbol{H}_2, \cdots, \boldsymbol{H}_{k-1}$ 使

$$\boldsymbol{H}_{k-1}\cdots\boldsymbol{H}_2\boldsymbol{H}_1\boldsymbol{A} = \boldsymbol{A}^{(k)},$$

其中

$$\boldsymbol{A}^{(k)} = \begin{pmatrix} -\sigma_1 & a_{12}^{(2)} & \cdots & a_{1k-1}^{(2)} & a_{1k}^{(2)} & \cdots & a_{1n}^{(2)} \\ & -\sigma_2 & \cdots & a_{2k-1}^{(3)} & a_{2k}^{(3)} & & a_{2n}^{(3)} \\ & & \ddots & \vdots & \vdots & & \vdots \\ & & & -\sigma_{k-1} & a_{k-1k}^{(k-1)} & \cdots & a_{k-1n}^{(k-1)} \\ & & & & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} \\ & & & & \vdots & & \vdots \\ & & & & a_{mk}^{(k)} & \cdots & a_{mn}^{(k)} \end{pmatrix}$$

$$= \begin{pmatrix} \boldsymbol{R}_k & \boldsymbol{r}_k & \boldsymbol{B}_k \\ \boldsymbol{0} & \boldsymbol{c}_k & \boldsymbol{D}_k \end{pmatrix} \begin{matrix} k-1 \\ m-k+1 \end{matrix},$$

其中，$\boldsymbol{R}_k$ 为 $k-1$ 阶上三角阵，$\boldsymbol{c}_k \in \mathbf{R}^{m-k+1}$，$\boldsymbol{D}_k \in \mathbf{R}^{(m-k+1)\times(n-k)}$.

不妨设 $\boldsymbol{c}_k \neq \boldsymbol{0}$，否则这一步不需要约化（如果 $\boldsymbol{A}$ 列满秩，则 $\boldsymbol{c}_k \neq \boldsymbol{0}$）. 于是，可选取初等反射阵 $\boldsymbol{H}'_k = \boldsymbol{I} - \beta_k^{-1}\boldsymbol{u}'_k\boldsymbol{u}'^{\mathrm{T}}_k$ 使

$$\boldsymbol{H}'_k\boldsymbol{c}_k = -\sigma_k\boldsymbol{e}_1.$$

令

$$\boldsymbol{H}_k=\begin{pmatrix}\boldsymbol{I}_{k-1} & \\ & \boldsymbol{H}_k'\end{pmatrix}\begin{matrix}k-1\\ m-k+1\end{matrix},$$

第 k 步约化为

$$\begin{aligned}\boldsymbol{H}_k\boldsymbol{A}^{(k)}&=\boldsymbol{H}_k\boldsymbol{H}_{k-1}\cdots\boldsymbol{H}_2\boldsymbol{H}_1\boldsymbol{A}=\boldsymbol{A}^{(k+1)}\\ &=\begin{pmatrix}\boldsymbol{I}_{k-1} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{H}_k'\end{pmatrix}\begin{pmatrix}\boldsymbol{R}_k & \boldsymbol{r}_k & \boldsymbol{B}_k\\ \boldsymbol{0} & \boldsymbol{c}_k & \boldsymbol{D}_k\end{pmatrix}\\ &=\begin{pmatrix}\boldsymbol{R}_k & \boldsymbol{r}_k & \boldsymbol{B}_k\\ \boldsymbol{0} & \boldsymbol{H}_k'\boldsymbol{c}_k & \boldsymbol{H}_k'\boldsymbol{D}_k\end{pmatrix},\end{aligned}$$

其中 $\mathbf{A}^{(k+1)}$ 左上角 k 阶子矩阵为 k 阶上三角阵 $\mathbf{R}_{k+1}$，这就使 $\boldsymbol{A}$ 三角化过程前进了一步.

令 $s=\min(m-1,n)$，继续上述过程，最后有

$$\boldsymbol{H}_s\cdots\boldsymbol{H}_2\boldsymbol{H}_1\boldsymbol{A}=\boldsymbol{A}^{(s+1)}=\boldsymbol{R}(\text{上梯形}).$$

第 k 步需要计算(σ_k, $\boldsymbol{u}_k'$, β_k)及 $\boldsymbol{H}_k'\boldsymbol{D}_k$，第 k 步约化大约需要 $2\cdot(m-k+1)(n-k)$ 次乘法运算.

总结上述讨论给出下述结果.

定理 29(矩阵的正交约化) 设 $\boldsymbol{A}\in\mathbf{R}^{m\times n}$ 且 $\boldsymbol{A}\neq\boldsymbol{0}$, $s=\min(m-1,n)$，则存在初等反射阵 $\boldsymbol{H}_1,\boldsymbol{H}_2,\cdots,\boldsymbol{H}_s$ 使

$$\boldsymbol{H}_s\cdots\boldsymbol{H}_2\boldsymbol{H}_1\boldsymbol{A}=\boldsymbol{R}(\text{上梯形}),$$

且计算量约为 $n^2m-n^3/3$(当 $m\geqslant n$)次乘法运算.

定理 30(矩阵的 QR 分解)

(1) 设 $\boldsymbol{A}\in\mathbf{R}^{m\times n}$ 且 $\boldsymbol{A}$ 的秩为 $n(m>n)$，则存在初等反射阵 $\boldsymbol{H}_1,\boldsymbol{H}_2,\cdots,\boldsymbol{H}_n$ 使

$$\boldsymbol{H}_n\cdots\boldsymbol{H}_2\boldsymbol{H}_1\boldsymbol{A}=\begin{pmatrix}\boldsymbol{R}\\ \boldsymbol{0}\end{pmatrix},$$

其中 $\boldsymbol{R}$ 为 n 阶非奇异上三角阵.

(2) 设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 为非奇异矩阵，则 $\boldsymbol{A}$ 有分解

$$\boldsymbol{A}=\boldsymbol{Q}\boldsymbol{R},$$

其中 Q 为正交矩阵，R 为上三角矩阵. 且当 R 具有正对角元时，分解唯一.

证明 (1) 由定理 29 可得.

(2) 由设及定理 29 存在初等反射阵 $H_1, H_2, \cdots, H_{n-1}$ 使

$$H_{n-1}\cdots H_2 H_1 A = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ & r_{22} & \cdots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \end{pmatrix} \equiv R.$$

记 $Q^{\mathrm{T}} = H_{n-1}\cdots H_2 H_1$，则上式为

$$Q^{\mathrm{T}} A = R,$$

即

$$A = QR,$$

其中 Q 为正交矩阵，R 为上三角阵.

唯一性. 设有 $A = Q_1 R_1 = Q_2 R_2$，其中 Q_1, Q_2 为正交阵，R_1, R_2 为非奇异上三角阵，且 R_1, R_2 具有正对角元素，则

$$A^{\mathrm{T}} A = R_1^{\mathrm{T}} Q_1^{\mathrm{T}} Q_1 R_1 = R_1^{\mathrm{T}} R_1,$$

$$A^{\mathrm{T}} A = R_2^{\mathrm{T}} Q_2^{\mathrm{T}} Q_2 R_2 = R_2^{\mathrm{T}} R_2.$$

由假设及对称正定矩阵 $A^{\mathrm{T}} A$ 的 Cholesky 分解的唯一性，则 $R_1 = R_2$. 从而可得 $Q_1 = Q_2$.

下面考虑用平面旋转变换来约化矩阵.

定理 31(用吉文斯变换计算矩阵的 QR 分解) 设 $A \in \mathbf{R}^{n\times n}$ 为非奇异矩阵，则

(1) 存在正交矩阵 $P_1, P_2, \cdots, P_{n-1}$ 使

$$P_{n-1}\cdots P_2 P_1 A = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ & r_{22} & \cdots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \end{pmatrix} \equiv R.$$

(2) A 有 QR 分解

$$\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{R},$$

其中 $\boldsymbol{Q}$ 为正交阵，$\boldsymbol{R}$ 为非奇异上三角阵，且当 $\boldsymbol{R}$ 对角元素都为正时，分解是唯一的.

证明　(1) 第 1 步约化：由设有 $j(1\leqslant j\leqslant n)$ 使 $a_{j1}\neq 0$. 若 $a_{j1}\neq 0(j=2,\cdots,n)$，则可选择吉文斯变换 $\boldsymbol{P}(1,j)(j=2,\cdots,n)$ 使

$$\boldsymbol{P}(1,n)\cdots\boldsymbol{P}(1,2)\boldsymbol{A} = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ & \vdots & & \vdots \\ & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} \end{pmatrix} \equiv \boldsymbol{A}^{(2)},$$

可简记为 $\boldsymbol{P}_1\boldsymbol{A}=\boldsymbol{A}^{(2)}$，其中 $\boldsymbol{P}_1=\boldsymbol{P}(1,n)\cdots\boldsymbol{P}(1,2)$.

(2) 第 k 步约化：设上述过程已完成第 1 步至第 $k-1$ 步，于是有

$$\boldsymbol{P}_{k-1}\cdots\boldsymbol{P}_2\boldsymbol{P}_1\boldsymbol{A} = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1k} & \cdots & r_{1n} \\ & r_{22} & \cdots & r_{2k} & \cdots & r_{2n} \\ & & \ddots & \vdots & & \vdots \\ & & & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} \\ & & & \vdots & & \vdots \\ & & & a_{nk}^{(k)} & \cdots & a_{nn}^{(k)} \end{pmatrix} \equiv \boldsymbol{A}^{(k)}.$$

由设有 $j(n\geqslant j\geqslant k)$ 使 $a_{jk}^{(k)}\neq 0$，若 $a_{jk}^{(k)}\neq 0(j=k+1,\cdots,n)$，则可选择吉文斯变换 $\boldsymbol{P}(k,j)(j=k+1,\cdots,n)$ 使

$$\begin{aligned} \boldsymbol{P}_k\boldsymbol{A}^{(k)} &= \boldsymbol{P}(k,n)\cdots\boldsymbol{P}(k,k+1)\boldsymbol{A}^{(k)} \\ &= \boldsymbol{P}_k\boldsymbol{P}_{k-1}\cdots\boldsymbol{P}_1\boldsymbol{A} = \boldsymbol{A}^{(k+1)}, \end{aligned}$$

其中 $\boldsymbol{P}_k\equiv\boldsymbol{P}(k,n)\cdots\boldsymbol{P}(k,k+1)$.

(3) 继续上述约化过程，最后则有

$$\boldsymbol{P}_{n-1}\cdots\boldsymbol{P}_2\boldsymbol{P}_1\boldsymbol{A} = \boldsymbol{R}\quad（上三角阵），$$

其中 $\boldsymbol{P}_k$ 为正交阵(为一系列平面旋转阵的乘积). 记 $\boldsymbol{Q}^{\mathrm{T}}=\boldsymbol{P}_{n-1}\cdots$

$\boldsymbol{P}_1$，则 $\boldsymbol{A}$ 有 QR 分解

$$\boldsymbol{A} = \boldsymbol{QR},$$

其中 $\boldsymbol{Q}$ 为正交矩阵，$\boldsymbol{R}$ 为非奇异上三角阵.

用 $\boldsymbol{H}$ 变换实现 $\boldsymbol{A}$ 的正交三角约化需要 $\frac{2}{3}n^3$ 次乘法，n 次开方运算，而用吉文斯变换约化计算约需要 $\frac{4}{3}n^3$ 次乘法，$\frac{n^2}{2}$ 次开方运算. 这说明，对一般矩阵 $\boldsymbol{A} \in \mathbf{R}^{n\times n}$ 用吉文斯变换实现 $\boldsymbol{A}$ 正交三角约化比用 $\boldsymbol{H}$ 变换实现 $\boldsymbol{A}$ 正交三角约化，计算量要大一倍. 但是，如果 $\boldsymbol{A}$ 为三对角阵或上豪斯霍尔德阵，利用吉文斯变换实现 $\boldsymbol{A}$ 正交三角约化要简单、经济.

5.7.4 求解超定方程组

设 $\boldsymbol{Ax} = \boldsymbol{b}$，其中 $\boldsymbol{A} \in \mathbf{R}^{m\times n}$. 当 $m > n$ 时称为超定方程组，一般地，没有通常意义下的解.

线性最小二乘问题：对超定方程组，寻求 $\boldsymbol{x}^* \in \mathbf{R}^n$ 使

$$\min_{x\in \mathbf{R}^n} \| \boldsymbol{b} - \boldsymbol{Ax} \|_2^2 = \| \boldsymbol{b} - \boldsymbol{Ax}^* \|_2^2.$$

如果使上式成立的 $\boldsymbol{x}^* \in \mathbf{R}^n$ 存在，称 $\boldsymbol{x}^*$ 为超定方程组最小二乘解.

记残差向量 $\boldsymbol{r} = \boldsymbol{b} - \boldsymbol{Ax}$，于是

$$\| \boldsymbol{r} \|_2^2 = \| \boldsymbol{b} - \boldsymbol{Ax} \|_2^2 = \sum_{i=1}^{m} \left(b_i - \sum_{j=1}^{n} a_{ij} x_j \right)^2,$$

即寻求 $\boldsymbol{x}^* = (x_1^*, \cdots, x_n^*)^{\mathrm{T}} \in \mathbf{R}^n$ 使偏差 r_i 的平方和最小.

设 $\boldsymbol{A} \in \mathbf{R}^{m\times n} (m > n)$ 且 $\boldsymbol{A}$ 具有线性无关的列，可利用 $\boldsymbol{A}$ 的正交约化来求超定方程组的最小二乘解.

由正交约化定理，可选择初等反射阵 $\boldsymbol{H}_1, \boldsymbol{H}_2, \cdots, \boldsymbol{H}_n$ 使

$$\boldsymbol{H}_n \cdots \boldsymbol{H}_2 \boldsymbol{H}_1 \boldsymbol{A} = \begin{pmatrix} \boldsymbol{R} \\ \boldsymbol{0} \end{pmatrix} \begin{matrix} n \\ m-n \end{matrix},$$

且同时约化常数项

$$H_n \cdots H_2 H_1 b = \begin{pmatrix} c \\ d \end{pmatrix} \begin{matrix} n \\ m-n \end{matrix},$$

其中,$R \in \mathbf{R}^{n\times n}$ 为非奇异矩阵,$c \in \mathbf{R}^n$.

令 $Q \equiv H_n \cdots H_2 H_1$,于是

$$Qr = Qb - QAx = \begin{pmatrix} c \\ d \end{pmatrix} - \begin{pmatrix} R \\ 0 \end{pmatrix} x = \begin{pmatrix} c - Rx \\ d \end{pmatrix}.$$

因为 Q 为正交矩阵,所以

$$\| r \|_2^2 = \| Qr \|_2^2 = \| c - Rx \|_2^2 + \| d \|_2^2 \geqslant \| d \|_2^2$$

$$(\text{对任何 } x \in \mathbf{R}^n).$$

从而,当选取 x^* 为 $Rx = c$ 的解时,则

$$\min_{x \in \mathbf{R}^n} \| r \|_2^2 = \| b - Ax^* \|_2^2 = \| r^* \|_2^2,$$

且达到极小的残差向量:

$$Qr^* = \begin{pmatrix} c - Rx^* \\ d \end{pmatrix} = \begin{pmatrix} 0 \\ d \end{pmatrix},$$

$$r^* = H_1 H_2 \cdots H_n \begin{pmatrix} 0 \\ d \end{pmatrix}.$$

算法 7(超定方程组)　设 $Ax = b$,其中 $A \in \mathbf{R}^{m\times n}$ $(m > n)$,且设 $\operatorname{rank}(A) = n$(列满秩),本算法利用正交三角约化 $H_n \cdots H_2 H_1 A = \begin{pmatrix} R \\ 0 \end{pmatrix}$,其中 $R \in \mathbf{R}^{n\times n}$ 为非奇异阵,求 $x^* \in \mathbf{R}^n$ 使 $\min_{x \in \mathbf{R}^n} \| b - Ax \|_2^2 = \| b - Ax^* \|_2^2$. 达到极小的残差向量冲掉 b.

(1) 正交约化

对于 $k = 1, 2, \cdots, n$

1) $A \leftarrow H_k A$　(即计算 $D_k \leftarrow H_k' D_k$)

2) $b \leftarrow H_k b$

(2) $c_i \leftarrow b_i$　$(i = 1, 2, \cdots, n)$

(3) 求解三角形方程组,求最小二乘解 x^*

$$Rx = c$$

(4) $b_i \leftarrow 0 \quad (i=1,2,\cdots,n)$

(5) 求达到极小的残差向量

对于 $k=n,n-1,\cdots,1$

$$b \leftarrow H_k b$$

算法 7 是解线性最小二乘问题的一个非常稳定的方法.

例 12 用正交约化方法求超定方程组

$$\begin{pmatrix} 2 & -1 \\ 8 & 4 \\ 2 & 1 \\ 7 & -1 \\ 4 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 8 \\ 3 \end{pmatrix}$$

的最小二乘解.

解 记$(A|b)\equiv(a_1\ a_2\ a_3)$, $A\in \mathbf{R}^{5\times 2}$, $\mathrm{rank}(A)=2$.

(1) 正交约化

(a) 利用算法 6,选择 $H_1=I-\beta'^{-1}_1 u'_1 u'^{\mathrm{T}}_1$, 使 $H_1 a_1=-\sigma_1 e_1$, 其中

$$\begin{cases} \sigma'_1 = 1.463087, \\ u'_1 = (1.713087,\ 1,\ 0.25,\ 0.875,\ 0.5)^{\mathrm{T}}, \\ \beta'_1 = \sigma'_1(0.25+\sigma'_1) = 2.506395, \\ \sigma_1 = 11.704696. \end{cases}$$

于是

$$H_1(A \mid b) = \begin{pmatrix} -11.704696 & -2.135895 & -6.151378 \\ & 3.336931 & -4.174556 \\ & 0.8342327 & -0.043639 \\ & -1.580185 & 4.347264 \\ & -0.331535 & 0.912722 \end{pmatrix}.$$

(b) 对于 $c_2 = (3.336931, 0.8342327, -1.580185, -0.331535)^T$ 利用算法 6，确定 $H'_2 = I - \beta_2'^{-1} u'_2 u_2'^T$ 使 $H'_2 c_2 = -\sigma_2 e_1$.

$$\begin{cases} \sigma'_2 = 1.138689, \\ u'_2 = (2.138689, 0.249999, -0.4735444, -0.09935326)^T, \\ \beta'_2 = 2.435302, \\ \sigma_2 = 3.799727. \end{cases}$$

于是

$$H_2 H_1 (A \mid b) = \begin{pmatrix} -11.704696 & -2.135895 & -6.151378 \\ 0 & -3.799727 & 5.563212 \\ & & 1.094643 \\ & & 2.191146 \\ & & 0.460352 \end{pmatrix}$$

$$= \begin{pmatrix} R & c \\ 0 & d \end{pmatrix}.$$

(2) 求解 $Rx = c$，即

$$\begin{pmatrix} -11.704696 & -2.135895 \\ 0 & -3.799727 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} -6.151378 \\ 5.563212 \end{pmatrix},$$

得到超定方程组的最小二乘解

$$x^* = \begin{pmatrix} 0.792721 \\ -1.46108 \end{pmatrix}.$$

评　　注

对于良态问题，高斯消去法也可能给出很坏的结果，即说明这个算法是不稳定的. 在高斯消去法中引进选主元的技巧，就得到了解方程组的完全主元素消去法和列主元素消去法，选主元素技

巧的根本作用是为了对增长因子(即 $r=\max\limits_{1\leqslant k\leqslant n}|a_k|/a$,其中 $a_k=\max\limits_{1\leqslant i,j\leqslant n}|a_{ij}^{(k)}|$,$a=\max\limits_{1\leqslant i,j\leqslant n}|a_{ij}|$,$\boldsymbol{A}^{(k)}=(a_{ij}^{(k)})$)进行控制,即对舍入误差的增长加以控制. 由此,完全选主素消去法及列主元消去法是数值稳定的算法. 这两种方法都是计算机上解线性方程组的有效方法. 但通常用列主元消去法即可.

从代数上看,直接分解法和高斯消去法本质上一样,但如果我们采用“双精度累加”计算 $\sum a_i b_i$,那么直接三角分解法的精度要比高斯消去法为高.

对于对称正定方程组,采用不选主元素的平方根法(或改进的平方根法)求解适宜. 理论分析指出,解对称正定方程组的平方根法是一个稳定的算法,在工程计算中使用比较广泛.

追赶法是解三对角线方程组(对角元占优势)的有效方法,它具有计算量少,方法简单,算法稳定等优点.

关于矩阵的条件数,病态方程组,算法的稳定性等都是计算数学中比较重要的概念,在这里我们只做了简单的介绍. 本章的内容可进一步参考文献[3],[7],[13].

习　题

1. 设 $\boldsymbol{A}$ 是对称阵且 $a_{11}\neq 0$,经过高斯消去法一步后,$\boldsymbol{A}$ 约化为

$$\begin{pmatrix} a_{11} & \boldsymbol{a}_1^{\mathrm{T}} \\ \boldsymbol{0} & \boldsymbol{A}_2 \end{pmatrix}.$$

证明 $\boldsymbol{A}_2$ 是对称矩阵.

2. 设 $\boldsymbol{A}=(a_{ij})_n$ 是对称正定矩阵,经过高斯消去法一步后,$\boldsymbol{A}$ 约化为

$$\begin{pmatrix} a_{11} & \boldsymbol{a}_1^{\mathrm{T}} \\ \boldsymbol{0} & \boldsymbol{A}_2 \end{pmatrix},$$

其中 $\boldsymbol{A}_2=(a_{ij}^{(2)})_{n-1}$. 证明:

(1) $\boldsymbol{A}$ 的对角元素 $a_{ii}>0(i=1,2,\cdots,n)$;

(2) $\boldsymbol{A}_2$ 是对称正定矩阵.

3. 设 $\boldsymbol{L}_k$ 为指标为 k 的初等下三角阵(除第 k 列对角元以下元素外,$\boldsymbol{L}_k$ 和单位阵 $\boldsymbol{I}$ 相同),即

$$\boldsymbol{L}_k=\begin{array}{c}\begin{array}{cccccc}&&k&&&\end{array}\\\begin{pmatrix}1&&&&&\\&\ddots&&&&\\&&1&&&\\&&m_{k+1,k}&1&&\\&&\vdots&&\ddots&\\&&m_{n,k}&&&1\end{pmatrix}\end{array}.$$

求证当 $i,j>k$ 时,$\widetilde{\boldsymbol{L}}_k=\boldsymbol{I}_{ij}\boldsymbol{L}_k\boldsymbol{I}_{ij}$ 也是一个指标为 k 的初等下三角阵,其中 $\boldsymbol{I}_{ij}$ 为初等置换阵.

4. 试推导矩阵 $\boldsymbol{A}$ 的 Crout 分解 $\boldsymbol{A}=\boldsymbol{LU}$ 的计算公式,其中 $\boldsymbol{L}$ 为下三角阵,$\boldsymbol{U}$ 为单位上三角阵.

5. 设 $\boldsymbol{Ux}=\boldsymbol{d}$,其中 $\boldsymbol{U}$ 为三角矩阵.

(a) 就 $\boldsymbol{U}$ 为上及下三角矩阵推导一般的求解公式,并写出算法.

(b) 计算解三角形方程组 $\boldsymbol{Ux}=\boldsymbol{d}$ 的乘除法次数.

(c) 设 $\boldsymbol{U}$ 为非奇异阵,试推导求 $\boldsymbol{U}^{-1}$ 的计算公式.

6. 证明:(a) 如果 $\boldsymbol{A}$ 是对称正定阵,则 $\boldsymbol{A}^{-1}$ 也是对称正定阵;

(b) 如果 $\boldsymbol{A}$ 是对称正定阵,则 $\boldsymbol{A}$ 可唯一地写成 $\boldsymbol{A}=\boldsymbol{L}^{\mathrm{T}}\boldsymbol{L}$,其中 $\boldsymbol{L}$ 是具有正对角元的下三角阵.

7. 用高斯-若当方法求 $\boldsymbol{A}$ 的逆矩阵,其中

$$\boldsymbol{A}=\begin{pmatrix}2&1&-3&-1\\3&1&0&7\\-1&2&4&-2\\1&0&-1&5\end{pmatrix}.$$

8. 用追赶法解三对角方程组 $\boldsymbol{Ax}=\boldsymbol{b}$,其中

$$A=\begin{pmatrix}2&-1&0&0&0\\-1&2&-1&0&0\\0&-1&2&-1&0\\0&0&-1&2&-1\\0&0&0&-1&2\end{pmatrix},\ b=\begin{pmatrix}1\\0\\0\\0\\0\end{pmatrix}.$$

9. 用改进的平方根法解方程组

$$\begin{pmatrix}2&-1&1\\-1&-2&3\\1&3&1\end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\end{pmatrix}=\begin{pmatrix}4\\5\\6\end{pmatrix}.$$

10. 下述矩阵能否分解为 LU(其中 L 为单位下三角阵,U 为上三角阵)?若能分解,那么分解是否唯一?

$$A=\begin{pmatrix}1&2&3\\2&4&1\\4&6&7\end{pmatrix},B=\begin{pmatrix}1&1&1\\2&2&1\\3&3&1\end{pmatrix},C=\begin{pmatrix}1&2&6\\2&5&15\\6&15&46\end{pmatrix}.$$

11. 设

$$A=\begin{pmatrix}0.6&0.5\\0.1&0.3\end{pmatrix},$$

计算 A 的行范数,列范数,2-范数及 F-范数.

12. 求证:(a) $\|x\|_\infty\leqslant\|x\|_1\leqslant n\|x\|_\infty$,

$$\text{(b)}\ \frac{1}{\sqrt{n}}\|A\|_F\leqslant\|A\|_2\leqslant\|A\|_F.$$

13. 设 $P\in\mathbf{R}^{n\times n}$ 且非奇异,又设 $\|x\|$ 为 $\mathbf{R}^n$ 上一向量范数,定义

$$\|x\|_P=\|Px\|.$$

试证明 $\|x\|_P$ 是 $\mathbf{R}^n$ 上向量的一种范数.

14. 设 $A\in\mathbf{R}^{n\times n}$ 为对称正定阵,定义

$$\|x\|_A=(Ax,x)^{\frac{1}{2}},$$

试证明 $\|x\|_A$ 为 $\mathbf{R}^n$ 上向量的一种范数.

15. 设 $\|A\|_s$, $\|A\|_t$ 为 $\mathbf{R}^{n\times n}$ 上任意两种矩阵算子范数,证明存在常数 $c_1,c_2>0$,使对一切 $A\in\mathbf{R}^{n\times n}$ 满足

$$c_1\|A\|_s\leqslant\|A\|_t\leqslant c_2\|A\|_s.$$

16. 设 $\boldsymbol{A}$ 为非奇异矩阵,求证

$$\frac{1}{\|\boldsymbol{A}^{-1}\|_\infty} = \min_{y\neq 0} \frac{\|\boldsymbol{A}\boldsymbol{y}\|_\infty}{\|\boldsymbol{y}\|_\infty}.$$

17. 矩阵第一行乘以一数,成为

$$\boldsymbol{A} = \begin{pmatrix} 2\lambda & \lambda \\ 1 & 1 \end{pmatrix},$$

证明当 $\lambda = \pm\frac{2}{3}$ 时,$\mathrm{cond}(\boldsymbol{A})_\infty$ 有最小值.

18. 设

$$\boldsymbol{A} = \begin{pmatrix} 100 & 99 \\ 99 & 98 \end{pmatrix},$$

计算 $\boldsymbol{A}$ 的条件数 $\mathrm{cond}(\boldsymbol{A})_v\ (v=2,\ \infty)$.

19. 证明:如果 $\boldsymbol{A}$ 是正交阵,则 $\mathrm{cond}(\boldsymbol{A})_2 = 1$.

20. 设 $\boldsymbol{A},\boldsymbol{B}\in\mathbf{R}^{n\times n}$,且 $\|\cdot\|$ 为 $\mathbf{R}^{n\times n}$ 上矩阵的算子范数,证明:

$$\mathrm{cond}(\boldsymbol{A}\boldsymbol{B}) \leqslant \mathrm{cond}(\boldsymbol{A})\mathrm{cond}(\boldsymbol{B}).$$

21. 设 $\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}$,其中 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 为非奇异阵,证明:

(a) $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$ 为对称正定矩阵;

(b) $\mathrm{cond}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})_2 = [\mathrm{cond}(\boldsymbol{A})_2]^2$.

第 6 章　解线性方程组的迭代法

6.1 引　　言

考虑线性方程组

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \tag{1.1}$$

其中 $\boldsymbol{A}$ 为非奇异矩阵，当 $\boldsymbol{A}$ 为低阶稠密矩阵时，第 5 章所讨论的选主元消去法是解(1.1)的有效方法. 但是，对于由工程技术中产生的大型稀疏矩阵方程组($\boldsymbol{A}$ 的阶数 n 很大，但零元素较多，例如求某些偏微分方程数值解所产生的线性方程组，$n \geqslant 10^4$)，利用迭代法求解(1.1)是合适的. 在计算机内存和运算两方面，迭代法通常都可利用 $\boldsymbol{A}$ 中有大量零元素的特点.

本章将介绍迭代法的一些基本理论及雅可比迭代法，高斯-塞德尔迭代法，超松弛迭代法，而超松弛迭代法应用很广泛.

下面举简例，以便了解迭代法的思想.

例 1　求解方程组

$$\begin{cases} 8x_1 - 3x_2 + 2x_3 = 20, \\ 4x_1 + 11x_2 - x_3 = 33, \\ 6x_1 + 3x_2 + 12x_3 = 36. \end{cases} \tag{1.2}$$

记为 $\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}$，其中

$$\boldsymbol{A} = \begin{pmatrix} 8 & -3 & 2 \\ 4 & 11 & -1 \\ 6 & 3 & 12 \end{pmatrix}, \quad \boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}, \quad \boldsymbol{b} = \begin{pmatrix} 20 \\ 33 \\ 36 \end{pmatrix}.$$

方程组的精确解是 $\boldsymbol{x}^* = (3,2,1)^{\mathrm{T}}$. 现将(1.2)改写为

$$\begin{cases} x_1 = \frac{1}{8}(3x_2 - 2x_3 + 20), \\ x_2 = \frac{1}{11}(-4x_1 + x_3 + 33), \\ x_3 = \frac{1}{12}(-6x_1 - 3x_2 + 36). \end{cases} \tag{1.3}$$

或写为 $\boldsymbol{x}=\boldsymbol{B}_0\boldsymbol{x}+\boldsymbol{f}$,其中

$$\boldsymbol{B}_0 = \begin{bmatrix} 0 & \frac{3}{8} & -\frac{2}{8} \\ -\frac{4}{11} & 0 & \frac{1}{11} \\ -\frac{6}{12} & -\frac{3}{12} & 0 \end{bmatrix}, \quad \boldsymbol{f} = \begin{bmatrix} \frac{20}{8} \\ \frac{33}{11} \\ \frac{36}{12} \end{bmatrix}.$$

我们任取初始值,例如取 $\boldsymbol{x}^{(0)}=(0,0,0)^{\mathrm{T}}$. 将这些值代入(1.3)式右边(若(1.3)式为等式即求得方程组的解,但一般不满足),得到新的值 $\boldsymbol{x}^{(1)}=(x_1^{(1)},x_2^{(1)},x_3^{(1)})^{\mathrm{T}}=(2.5,3,3)^{\mathrm{T}}$,再将 $\boldsymbol{x}^{(1)}$ 分量代入(1.3)式右边得到 $\boldsymbol{x}^{(2)}$,反复利用这个计算程序,得到一向量序列和一般的计算公式(迭代公式)

$$\boldsymbol{x}^{(0)} = \begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \\ x_3^{(0)} \end{bmatrix}, \ \boldsymbol{x}^{(1)} = \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \\ x_3^{(1)} \end{bmatrix}, \ \cdots, \ \boldsymbol{x}^{(k)} = \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ x_3^{(k)} \end{bmatrix}, \ \cdots$$

$$\begin{cases} x_1^{(k+1)} = (3x_2^{(k)} - 2x_3^{(k)} + 20)/8, \\ x_2^{(k+1)} = (-4x_1^{(k)} + x_3^{(k)} + 33)/11, \\ x_3^{(k+1)} = (-6x_1^{(k)} - 3x_2^{(k)} + 36)/12. \end{cases} \tag{1.4}$$

简写为

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{B}_0\boldsymbol{x}^{(k)} + \boldsymbol{f},$$

其中 k 表示迭代次数($k=0,1,2,\cdots$).

迭代到第 10 次有

$$x^{(10)} = (3.000032, 1.999838, 0.9998813)^T;$$

$$\| \boldsymbol{\varepsilon}^{(10)} \|_\infty = 0.000187(\boldsymbol{\varepsilon}^{(10)} = \boldsymbol{x}^{(10)} - \boldsymbol{x}^*).$$

从此例看出，由迭代法产生的向量序列 $\boldsymbol{x}^{(k)}$ 逐步逼近方程组的精确解 $\boldsymbol{x}^*$.

对于任何一个方程组 $\boldsymbol{x}=\boldsymbol{B}\boldsymbol{x}+\boldsymbol{f}$（由 $\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}$ 变形得到的等价方程组），由迭代法产生的向量序列 $\boldsymbol{x}^{(k)}$ 是否一定逐步逼近方程组的解 $\boldsymbol{x}^*$ 呢？回答是不一定. 请读者考虑用迭代法解下述方程组

$$\begin{cases} x_1 = 2x_2 + 5, \\ x_2 = 3x_1 + 5. \end{cases}$$

对于给定方程组 $\boldsymbol{x}=\boldsymbol{B}\boldsymbol{x}+\boldsymbol{f}$，设有唯一解 $\boldsymbol{x}^*$，则

$$\boldsymbol{x}^* = \boldsymbol{B}\boldsymbol{x}^* + \boldsymbol{f}. \tag{1.5}$$

又设 $\boldsymbol{x}^{(0)}$ 为任取的初始向量，按下述公式构造向量序列

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{B}\boldsymbol{x}^{(k)} + \boldsymbol{f},\ k = 0,1,2,\cdots, \tag{1.6}$$

其中 k 表迭代次数.

定义 1 (1) 对于给定的方程组 $\boldsymbol{x}=\boldsymbol{B}\boldsymbol{x}+\boldsymbol{f}$，用公式(1.6)逐步代入求近似解的方法称为**迭代法**（或称为一阶定常迭代法，这里 $\boldsymbol{B}$ 与 k 无关）.

(2) 如果 $\lim\limits_{k\to\infty}\boldsymbol{x}^{(k)}$ 存在（记为 $\boldsymbol{x}^*$），称此**迭代法收敛**，显然 $\boldsymbol{x}^*$ 就是方程组的解，否则称此**迭代法发散**.

由上述讨论，需要研究 $\{\boldsymbol{x}^{(k)}\}$ 的收敛性. 引进误差向量

$$\boldsymbol{\varepsilon}^{(k+1)} = \boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*,$$

由(1.6)减去(1.5)式，得 $\boldsymbol{\varepsilon}^{(k+1)}=\boldsymbol{B}\boldsymbol{\varepsilon}^{(k)}\ (k=0,1,2,\cdots)$，递推得

$$\boldsymbol{\varepsilon}^{(k)} = \boldsymbol{B}\boldsymbol{\varepsilon}^{(k-1)} = \cdots = \boldsymbol{B}^k\boldsymbol{\varepsilon}^{(0)}.$$

要考察 $\{\boldsymbol{x}^{(k)}\}$ 的收敛性. 就要研究 $\boldsymbol{B}$ 在什么条件下有 $\lim\limits_{k\to\infty}\boldsymbol{\varepsilon}^{(k)}=\boldsymbol{0}$，亦即要研究 $\boldsymbol{B}$ 满足什么条件时有 $\boldsymbol{B}^k\to\boldsymbol{0}$（零矩阵）$(k\to\infty)$.

6.2 基本迭代法

设有

$$\boldsymbol{Ax} = \boldsymbol{b}, \tag{2.1}$$

其中，$\boldsymbol{A}=(a_{ij})\in\mathbf{R}^{n\times n}$为非奇异矩阵. 下面研究如何建立解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的各种迭代法.

将 $\boldsymbol{A}$ 分裂为

$$\boldsymbol{A} = \boldsymbol{M} - \boldsymbol{N}, \tag{2.2}$$

其中，$\boldsymbol{M}$ 为可选择的非奇异矩阵，且使 $\boldsymbol{Mx}=\boldsymbol{d}$ 容易求解，一般选择为 $\boldsymbol{A}$ 的某种近似，称 $\boldsymbol{M}$ 为**分裂矩阵**.

于是，求解 $\boldsymbol{Ax}=\boldsymbol{b}$ 转化为求解 $\boldsymbol{Mx}=\boldsymbol{Nx}+\boldsymbol{b}$，即求解

$$\boldsymbol{Ax} = \boldsymbol{b} \Leftrightarrow \text{求解 } \boldsymbol{x} = \boldsymbol{M}^{-1}\boldsymbol{Nx} + \boldsymbol{M}^{-1}\boldsymbol{b}.$$

可构造一阶定常迭代法

$$\begin{cases} \boldsymbol{x}^{(0)}\ (\text{初始向量}), \\ \boldsymbol{x}^{(k+1)} = \boldsymbol{Bx}^{(k)} + \boldsymbol{f} \quad (k = 0,1,\cdots,), \end{cases} \tag{2.3}$$

其中 $\boldsymbol{B}=\boldsymbol{M}^{-1}\boldsymbol{N}=\boldsymbol{M}^{-1}(\boldsymbol{M}-\boldsymbol{A})=\boldsymbol{I}-\boldsymbol{M}^{-1}\boldsymbol{A}$，$\boldsymbol{f}=\boldsymbol{M}^{-1}\boldsymbol{b}$. 称 $\boldsymbol{B}=\boldsymbol{I}-\boldsymbol{M}^{-1}\boldsymbol{A}$ 为迭代法的迭代矩阵，选取 $\boldsymbol{M}$ 阵，就得到解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的各种迭代法.

设 $a_{ii}\neq 0(i=1,2,\cdots,n)$，并将 $\boldsymbol{A}$ 写为三部分

$$\boldsymbol{A} = \begin{pmatrix} a_{11} & & & \\ & a_{22} & & \\ & & \ddots & \\ & & & a_{nn} \end{pmatrix} - \begin{pmatrix} 0 & & & & \\ -a_{21} & 0 & & & \\ \vdots & \vdots & \ddots & & \\ -a_{n-1,1} & -a_{n-1,2} & \cdots & 0 & \\ -a_{n1} & -a_{n2} & \cdots & -a_{n,n-1} & 0 \end{pmatrix}$$

$$
-\begin{pmatrix} 0 & -a_{12} & \cdots & -a_{1,n-1} & -a_{1n} \\ & 0 & \cdots & -a_{2,n-1} & -a_{2n} \\ & & \ddots & \vdots & \vdots \\ & & & 0 & -a_{n-1,n} \\ & & & & 0 \end{pmatrix}
$$

$$
\equiv \boldsymbol{D}-\boldsymbol{L}-\boldsymbol{U}. \tag{2.4}
$$

6.2.1 雅可比迭代法

由 $a_{ii}\neq 0(i=1,\cdots,n)$，选取 $\boldsymbol{M}$ 为 $\boldsymbol{A}$ 的对角元素部分，即选取 $\boldsymbol{M}=\boldsymbol{D}$(对角阵)，$\boldsymbol{A}=\boldsymbol{D}-\boldsymbol{N}$，由(2.3)式得到解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的雅可比(Jacobi)迭代法

$$
\begin{cases} \boldsymbol{x}^{(0)} & (\text{初始向量}) \\ \boldsymbol{x}^{(k+1)}=\boldsymbol{Bx}^{(k)}+\boldsymbol{f} & (k=0,1,\cdots), \end{cases} \tag{2.5}
$$

其中 $\boldsymbol{B}=\boldsymbol{I}-\boldsymbol{D}^{-1}\boldsymbol{A}=\boldsymbol{D}^{-1}(\boldsymbol{L}+\boldsymbol{U})\equiv\boldsymbol{J}$, $\boldsymbol{f}=\boldsymbol{D}^{-1}\boldsymbol{b}$. 称 $\boldsymbol{J}$ 为解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的雅可比迭代法的迭代阵.

下面给出雅可比迭代法(2.5)的分量计算公式，记

$$
\boldsymbol{x}^{(k)}=(x_1^{(k)},\cdots,x_i^{(k)},\cdots,x_n^{(k)})^{\mathrm{T}},
$$

由雅可比迭代公式(2.5)有

$$
\boldsymbol{Dx}^{(k+1)}=(\boldsymbol{L}+\boldsymbol{U})\boldsymbol{x}^{(k)}+\boldsymbol{b},
$$

或 $a_{ii}x_i^{(k+1)}=-\sum_{j=1}^{i-1}a_{ij}x_j^{(k)}-\sum_{j=i+1}^{n}a_{ij}x_j^{(k)}+b_i \quad (i=1,2,\cdots,n).$

于是，解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的雅可比迭代法的计算公式为

$$
\begin{cases} \boldsymbol{x}^{(0)}=(x_1^{(0)},\cdots,x_n^{(0)})^{\mathrm{T}}, \\ x_i^{(k+1)}=\left(b_i-\sum\limits_{\substack{j=1\\ j\neq i}}^{n}a_{ij}x_j^{(k)}\right)\Big/a_{ii} \\ (i=1,2,\cdots,n) \quad (k=0,1,\cdots \text{表示迭代次数}). \end{cases} \tag{2.6}
$$

由(2.6)式可知，雅可比迭代法计算公式简单，每迭代一次只需计算一次矩阵和向量的乘法且计算过程中原始矩阵 $\boldsymbol{A}$ 始终不变.

6.2.2 高斯-塞德尔迭代法

选取分裂矩阵 $\boldsymbol{M}$ 为 $\boldsymbol{A}$ 的下三角部分，即选取 $\boldsymbol{M}=\boldsymbol{D}-\boldsymbol{L}$(下三角阵)，$\boldsymbol{A}=\boldsymbol{M}-\boldsymbol{N}$，于是由(2.3)式得到解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的高斯-塞德尔(Gauss-Seidel)迭代法

$$\begin{cases} \boldsymbol{x}^{(0)} & \text{(初始向量)} \\ \boldsymbol{x}^{(k+1)} = \boldsymbol{Bx}^{(k)} + \boldsymbol{f} & (k = 0,1,\cdots), \end{cases} \tag{2.7}$$

其中 $\boldsymbol{B}=\boldsymbol{I}-(\boldsymbol{D}-\boldsymbol{L})^{-1}\boldsymbol{A}=(\boldsymbol{D}-\boldsymbol{L})^{-1}\boldsymbol{U}\equiv\boldsymbol{G}$，$\boldsymbol{f}=(\boldsymbol{D}-\boldsymbol{L})^{-1}\boldsymbol{b}$. 称 $\boldsymbol{G}=(\boldsymbol{D}-\boldsymbol{L})^{-1}\boldsymbol{U}$ 为解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的高斯-塞德尔迭代法的迭代阵.

下面给出高斯-塞德尔迭代法的分量计算公式. 记

$$\boldsymbol{x}^{(k)} = (x_1^{(k)},\cdots,x_i^{(k)},\cdots,x_n^{(k)})^{\mathrm{T}}.$$

由(2.7)式有

$$(\boldsymbol{D}-\boldsymbol{L})\boldsymbol{x}^{(k+1)} = \boldsymbol{Ux}^{(k)} + \boldsymbol{b},$$

或

$$\boldsymbol{Dx}^{(k+1)} = \boldsymbol{Lx}^{(k+1)} + \boldsymbol{Ux}^{(k)} + \boldsymbol{b},$$

即 $a_{ii}x_i^{(k+1)} = b_i - \sum\limits_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum\limits_{j=i+1}^{n} a_{ij}x_j^{(k)} \quad (i = 1,2,\cdots,n).$

于是解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的高斯-塞德尔迭代法计算公式为

$$\begin{cases} \boldsymbol{x}^{(0)} = (x_1^{(0)},\cdots,x_n^{(0)})^{\mathrm{T}} \quad \text{(初始向量)}, \\ x_i^{(k+1)} = \left(b_i - \sum\limits_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum\limits_{j=i+1}^{n} a_{ij}x_j^{(k)}\right)\Big/a_{ii} \\ (i = 1,\cdots,n;\ k = 0,1,\cdots). \end{cases} \tag{2.8}$$

或

$$
\begin{cases}
\boldsymbol{x}^{(0)} = (x_1^{(0)},\cdots,x_n^{(0)})^{\mathrm{T}}, \\
x_i^{(k+1)} = x_i^{(k)} + \Delta x_i \\
\Delta x_i = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij} x_j^{(k)}\right) \Big/ a_{ii} \\
(i = 1,\cdots,n;\ k = 0,1,\cdots).
\end{cases}
\tag{2.9}
$$

雅可比迭代法不使用变量的最新信息计算 $x_i^{(k+1)}$，而由高斯-塞德尔迭代公式(2.8)可知，计算 $\boldsymbol{x}^{(k+1)}$ 的第 i 个分量 $x_i^{(k+1)}$ 时，利用了已经计算出的最新分量 $x_j^{(k+1)}(j=1,2,\cdots,i-1)$. 高斯-塞德尔迭代法可看作雅可比迭代法的一种改进. 由(2.8)可知，高斯-塞德尔迭代法每迭代一次只需计算一次矩阵与向量的乘法.

算法 1(高斯-塞德尔迭代法) 设 $\boldsymbol{Ax}=\boldsymbol{b}$，其中 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 为非奇异矩阵且 $a_{ii}\neq 0(i=1,\cdots,n)$，本算法用高斯-塞德尔迭代法解 $\boldsymbol{Ax}=\boldsymbol{b}$，数组 $x(n)$ 开始存放 $\boldsymbol{x}^{(0)}$，后存放 $\boldsymbol{x}^{(k)}$，N_0 为最大迭代次数.

1. $x_i \leftarrow 0.0(i=1,\cdots,n)$
2. 对于 $k=1,2,\cdots,N_0$

对于 $i=1,2,\cdots,n$

$$
x_i \leftarrow \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j - \sum_{j=i+1}^{n} a_{ij} x_j\right) \Big/ a_{ii}
$$

迭代一次，这个算法需要的运算次数至多与矩阵 $\boldsymbol{A}$ 的非零元素的个数一样多.

例 2 用高斯-塞德尔迭代法解线性方程组(1.2).

高斯-塞德尔迭代公式：取 $\boldsymbol{x}^{(0)}=(0,0,0)^{\mathrm{T}}$.

$$
\begin{cases}
x_1^{(k+1)} = (20 + x_2^{(k)} - 2x_3^{(k)})/8, \\
x_2^{(k+1)} = (33 - 4x_1^{(k+1)} + x_3^{(k)})/11, \\
x_3^{(k+1)} = (36 - 6x_1^{(k+1)} - 3x_2^{(k+1)})/12.
\end{cases}
$$

$$(k = 0,1,\cdots,)$$

计算 $x^{(7)}=(3.000002,\ 1.9999987,\ 0.9999932)^{\mathrm{T}}$，且

$$\| x^* - x^{(7)} \|_{\infty} < 2.02 \times 10^{-6}.$$

由此例可知，用高斯-塞德尔迭代法，雅可比迭代法解线性方程组(1.2)(且取 $x^{(0)}=\mathbf{0}$)均收敛，而高斯-塞德尔迭代法比雅可比迭代法收敛较快(即取 $x^{(0)}$ 相同，达到同样精度所需迭代次数较少)，但这结论只当 A 满足一定条件时才是对的.

6.2.3 解大型稀疏线性方程组的逐次超松弛迭代法

选取分裂矩阵 M 为带参数的下三角阵

$$M = \frac{1}{\omega}(D - \omega L),$$

其中 $\omega>0$ 为可选择的松弛因子.

于是，由(2.3)可构造一个迭代法，其迭代矩阵为

$$\begin{aligned} L_{\omega} &\equiv I - \omega(D - \omega L)^{-1}A \\ &= (D - \omega L)^{-1}((1-\omega)D + \omega U). \end{aligned}$$

从而得到解 $Ax = b$ 的逐次超松弛迭代法(Successive Over Relaxation Method，简称 SOR 方法).

解 $Ax=b$ 的 SOR 方法为

$$\begin{cases} x^{(0)} & (\text{初始向量}) \\ x^{(k+1)} = L_{\omega}x^{(k)} + f & (k = 0,1,\cdots), \end{cases} \tag{2.10}$$

其中 $L_{\omega}=(D-\omega L)^{-1}((1-\omega)D+\omega U)$，$f=\omega(D-\omega L)^{-1}b$.

下面给出解 $Ax=b$ 的 SOR 迭代法的分量计算公式. 记

$$x^{(k)} = (x_1^{(k)},\cdots,x_i^{(k)},\cdots,x_n^{(k)})^{\mathrm{T}},$$

由(2.10)式可得

$$(D - \omega L)x^{(k+1)} = ((1-\omega)D + \omega U)x^{(k)} + \omega b,$$

或

$$Dx^{(k+1)} = Dx^{(k)} + \omega(b + Lx^{(k+1)} + Ux^{(k)} - Dx^{(k)}).$$

由此，得到解 $Ax=b$ 的 SOR 方法的计算公式

$$\begin{cases} x^{(0)} = (x_1^{(0)},\cdots,x_n^{(0)})^{\mathrm{T}}, \\ x_i^{(k+1)} = x_i^{(k)} + \omega\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij}x_j^{(k)}\right)\Big/ a_{ii} \\ (i = 1,2,\cdots,n;\ k = 0,1), \\ \omega \text{ 为松弛因子}. \end{cases} \tag{2.11}$$

或

$$\begin{cases} x^{(0)} = (x_1^{(0)},\ \cdots,\ x_n^{(0)})^{\mathrm{T}}, \\ x_i^{(k+1)} = x_i^{(k)} + \Delta x_i, \\ \Delta x_i = \omega\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij}x_j^{(k)}\right)\Big/ a_{ii} \\ (i = 1,2,\cdots,n;\ k = 0,1,\cdots), \\ \omega \text{ 为松弛因子}. \end{cases} \tag{2.12}$$

(1) 显然，当 $\omega=1$ 时，SOR 方法即为高斯-塞德尔迭代法.

(2) SOR 方法每迭代一次主要运算量是计算一次矩阵与向量的乘法.

(3) 当 $\omega>1$ 时，称为超松弛法；当 $\omega<1$ 时，称为低松弛法.

(4) 在计算机实现时可用

$$\max_{1\leqslant i\leqslant n} |\Delta x_i| = \max_{1\leqslant i\leqslant n} |x_i^{(k+1)} - x_i^{(k)}| < \varepsilon$$

控制迭代终止，或用 $\|r^{(k)}\|_\infty = \|b - Ax^{(k)}\|_\infty < \varepsilon$ 控制迭代终止.

SOR 迭代法是高斯-塞德尔迭代法的一种修正，可由下述思想得到.

设已知 $x^{(k)}$ 及已计算 $x^{(k+1)}$ 的分量 $x_j^{(k+1)}(j=1,2,\cdots,i-1)$.

(1) 首先用高斯-塞德尔迭代法定义辅助量 $\tilde{x}_i^{(k+1)}$,

$$\tilde{x}_i^{(k+1)} = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}\right) \Big/ a_{ii}. \tag{2.13}$$

(2) 再由 $x_i^{(k)}$ 与 $\tilde{x}_i^{(k+1)}$ 加权平均定义 $x_i^{(k+1)}$,即

$$x_i^{(k+1)} = (1-\omega) x_i^{(k)} + \omega \tilde{x}_i^{(k+1)} = x_i^{(k)} + \omega(\tilde{x}_i^{(k+1)} - x_i^{(k)}). \tag{2.14}$$

将(2.13)代入(2.14)得到解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的 SOR 迭代(2.11)式.

例 3 用 SOR 方法解方程组

$$\begin{pmatrix} -4 & 1 & 1 & 1 \\ 1 & -4 & 1 & 1 \\ 1 & 1 & -4 & 1 \\ 1 & 1 & 1 & -4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix},$$

它的精确解为 $\boldsymbol{x}^* = (-1, -1, -1, -1)^{\mathrm{T}}$.

解 取 $\boldsymbol{x}^{(0)} = \boldsymbol{0}$,迭代公式为

$$\begin{cases} x_1^{(k+1)} = x_1^{(k)} - \omega(1 + 4x_1^{(k)} - x_2^{(k)} - x_3^{(k)} - x_4^{(k)})/4; \\ x_2^{(k+1)} = x_2^{(k)} - \omega(1 - x_1^{(k+1)} + 4x_2^{(k)} - x_3^{(k)} - x_4^{(k)})/4; \\ x_3^{(k+1)} = x_3^{(k)} - \omega(1 - x_1^{(k+1)} - x_2^{(k+1)} + 4x_3^{(k)} - x_4^{(k)})/4; \\ x_4^{(k+1)} = x_4^{(k)} - \omega(1 - x_1^{(k+1)} - x_2^{(k+1)} - x_3^{(k+1)} + 4x_4^{(k)})/4. \end{cases}$$

取 $\omega = 1.3$,第 11 次迭代结果为

$$\boldsymbol{x}^{(11)} = (-0.99999646, -1.00000310, -0.99999953, -0.99999912)^{\mathrm{T}},$$

$$\|\boldsymbol{\varepsilon}^{(11)}\|_2 \leqslant 0.46 \times 10^{-5}.$$

对 ω 取其他值,迭代次数如下表. 从此例看到,松弛因子选择得好,会使 SOR 迭代法的收敛大大加速. 本例中 $\omega = 1.3$ 是最佳松弛因子.

表 6-1

松弛因子 ω	满足误差 $\Vert x^{(k)}-x^*\Vert_2<10^{-5}$ 的迭代次数	松弛因子 ω	满足误差 $\Vert x^{(k)}-x^*\Vert_2<10^{-5}$ 的迭代次数
1.0	22	1.5	17
1.1	17	1.6	23
1.2	12	1.7	33
1.3	11(最少迭代次数)	1.8	53
1.4	14	1.9	109

6.3 迭代法的收敛性

6.3.1 一阶定常迭代法的基本定理

设

$$\boldsymbol{Ax}=\boldsymbol{b}, \tag{3.1}$$

其中 $\mathbf{A}\in\mathbf{R}^{n\times n}$为非奇异矩阵，记 $\boldsymbol{x}^*$ 为(3.1)精确解，且设有等价的方程组

$$\boldsymbol{Ax}=\boldsymbol{b}\Leftrightarrow\boldsymbol{x}=\boldsymbol{Bx}+\boldsymbol{f}.$$

于是

$$\boldsymbol{x}^*=\boldsymbol{Bx}^*+\boldsymbol{f}. \tag{3.2}$$

设有解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的一阶定常迭代法

$$\boldsymbol{x}^{(k+1)}=\boldsymbol{Bx}^{(k)}+\boldsymbol{f}. \tag{3.3}$$

有意义的问题是：迭代矩阵 $\boldsymbol{B}$ 满足什么条件时，由迭代法产生的向量序列$\{\boldsymbol{x}^{(k)}\}$收敛到 $\boldsymbol{x}^*$.

引进误差向量

$$\boldsymbol{\varepsilon}^{(k)}=\boldsymbol{x}^{(k)}-\boldsymbol{x}^*\quad(k=0,1,2,\cdots).$$

由(3.3)式减(3.2)式得到误差向量的递推公式

$$\boldsymbol{\varepsilon}^{(k+1)} = \boldsymbol{B}\boldsymbol{\varepsilon}^{(k)},$$

$$\boldsymbol{\varepsilon}^{(k)} = \boldsymbol{B}^k\boldsymbol{\varepsilon}^{(0)} \quad (k = 0,1,\cdots).$$

由 6.1 节可知,研究迭代法(3.3)收敛性问题就是要研究迭代矩阵 $\boldsymbol{B}$ 满足什么条件时,有 $\boldsymbol{B}^k \to \boldsymbol{0}$(零矩阵)($k \to \infty$).

定义 2　设有矩阵序列 $\boldsymbol{A}_k = (a_{ij}^{(k)}) \in \mathbf{R}^{n\times n}$ 及 $\boldsymbol{A} = (a_{ij}) \in \mathbf{R}^{n\times n}$,如果 n^2 个数列极限存在且有

$$\lim_{k\to\infty} a_{ij}^{(k)} = a_{ij} \quad (i,j = 1,2,\cdots,n),$$

则称 $\{\boldsymbol{A}_k\}$ 收敛于 $\boldsymbol{A}$,记为 $\lim\limits_{k\to\infty}\boldsymbol{A}_k = \boldsymbol{A}$.

例 4　设有矩阵序列

$$\boldsymbol{A} = \begin{pmatrix} \lambda & 1 \\ 0 & \lambda \end{pmatrix},\ \boldsymbol{A}^2 = \begin{pmatrix} \lambda^2 & 2\lambda \\ 0 & \lambda^2 \end{pmatrix},\ \cdots,\ \boldsymbol{A}^k = \begin{pmatrix} \lambda^k & k\lambda^{k-1} \\ 0 & \lambda^k \end{pmatrix},\ \cdots$$

且设 $|\lambda| < 1$,考查其极限.

解　显然,当 $|\lambda| < 1$ 时,则有 $\lim\limits_{k\to\infty}\boldsymbol{A}_k \equiv \lim\limits_{k\to\infty}\boldsymbol{A}^k = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$.

矩阵序列极限概念可以用矩阵算子范数来描述.

定理 1　$\lim\limits_{k\to\infty}\boldsymbol{A}_k = \boldsymbol{A} \Leftrightarrow \lim\limits_{k\to\infty}\|\boldsymbol{A}_k - \boldsymbol{A}\| = 0$,其中 $\|\cdot\|$ 为矩阵的任意一种算子范数.

证明　显然有

$$\lim_{k\to\infty}\boldsymbol{A}_k = \boldsymbol{A} \Leftrightarrow \lim_{k\to\infty}\|\boldsymbol{A}_k - \boldsymbol{A}\|_\infty = 0.$$

再利用矩阵范数的等价性,可证定理对其他算子范数亦对.

定理 2　$\lim\limits_{k\to\infty}\boldsymbol{A}_k = \boldsymbol{A} \Leftrightarrow$ 是对任何向量 $\boldsymbol{x} \in \mathbf{R}^n$ 都有 $\lim\limits_{k\to\infty}\boldsymbol{A}_k\boldsymbol{x} = \boldsymbol{A}\boldsymbol{x}$.

证明作为练习.

定理 3　设 $\boldsymbol{B} = (b_{ij}) \in \mathbf{R}^{n\times n}$,则 $\lim\limits_{k\to\infty}\boldsymbol{B}^k = \boldsymbol{0}$(零矩阵)的充分必要条件是矩阵 $\boldsymbol{B}$ 的谱半径 $\rho(\boldsymbol{B}) < 1$.

证明　由矩阵 $\boldsymbol{B}$ 的若当标准型,存在非奇异矩阵 $\boldsymbol{P}$ 使

$$
\boldsymbol{P}^{-1}\boldsymbol{B}\boldsymbol{P} = \begin{pmatrix} \boldsymbol{J}_1 & & & \\ & \boldsymbol{J}_2 & & \\ & & \ddots & \\ & & & \boldsymbol{J}_r \end{pmatrix} \equiv \boldsymbol{J},
$$

其中若当块

$$
\boldsymbol{J}_i = \begin{pmatrix} \lambda_i & 1 & & \\ & \lambda_i & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_i \end{pmatrix}_{n_i \times n_i},
$$

且 $\sum_{i=1}^{r} n_i = n$，显然有

$$
\boldsymbol{B} = \boldsymbol{P}\boldsymbol{J}\boldsymbol{P}^{-1},
$$

$$
\boldsymbol{B}^k = \boldsymbol{P}\boldsymbol{J}^k\boldsymbol{P}^{-1},
$$

其中

$$
\boldsymbol{J}^k = \begin{pmatrix} \boldsymbol{J}_1^k & & & \\ & \boldsymbol{J}_2^k & & \\ & & \ddots & \\ & & & \boldsymbol{J}_r^k \end{pmatrix}.
$$

于是$\lim\limits_{k\to\infty}\boldsymbol{B}^k=\boldsymbol{0}\Leftrightarrow\lim\limits_{k\to\infty}\boldsymbol{J}^k=\boldsymbol{0}\Leftrightarrow\lim\limits_{k\to\infty}\boldsymbol{J}_i^k=\boldsymbol{0}(i=1,2,\cdots,r)$.

下面考查 $\boldsymbol{J}_i^k$ 的情况. 引进记号

$$
\boldsymbol{E}_{t,k} = \begin{pmatrix} \boldsymbol{0} & \boldsymbol{I} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix} \}t-k \in \mathbf{R}^{t\times t} \quad (t = n_i).
$$

显然有,$\boldsymbol{E}_{t,0}=\boldsymbol{I}$, $\boldsymbol{E}_{t,k}=\boldsymbol{0}$(当 $k\geqslant t$), $(\boldsymbol{E}_{t,1})^k=\boldsymbol{E}_{t,k}$. 由于 $\boldsymbol{J}_i=\lambda_i\boldsymbol{I}+\boldsymbol{E}_{t,1}$，因此

$$
\boldsymbol{J}_i^k = (\lambda_i\boldsymbol{I} + \boldsymbol{E}_{t,1})^k = \sum_{j=0}^{k} C_k^j \lambda_i^{k-j} (\boldsymbol{E}_{t,1})^j
$$

$$= \sum_{j=0}^{t-1} C_k^j \lambda_i^{k-j} E_{t,j}$$

$$= \begin{pmatrix} \lambda_i^k & C_k^1\lambda_i^{k-1} & C_k^2\lambda_i^{k-2} & \cdots & C_k^{t-1}\lambda_i^{k-(t-1)} \\ & \lambda_i^k & C_k^1\lambda_i^{k-1} & \cdots & C_k^{t-2}\lambda_i^{k-(t-2)} \\ & & \ddots & \ddots & \vdots \\ & & & \ddots & C_k^1\lambda_i^{k-1} \\ & & & & \lambda_i^k \end{pmatrix}_{t\times t}$$

$$(i = 1,2,\cdots,r),$$

其中

$$C_k^j = \frac{k!}{j!(k-j)!} = \frac{k(k-1)\cdots(k-j+1)}{j!}.$$

利用极限$\lim\limits_{k\to\infty} k^r c^k = 0(0<c<1, r\geqslant 0)$，得到

$$\lim_{k\to\infty} C_k^j \lambda^{k-j} = 0 \Leftrightarrow |\lambda| < 1.$$

所以$\lim\limits_{k\to\infty} \boldsymbol{J}_i^k = \boldsymbol{0}$ 的充要条件是$|\lambda_i|<1(i=1,2,\cdots,r)$，即 $\rho(\boldsymbol{B})<1$.

定理 4(迭代法基本定理) 设有方程组

$$\boldsymbol{x} = \boldsymbol{B}\boldsymbol{x} + \boldsymbol{f}, \tag{3.4}$$

及一阶定常迭代法

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{B}\boldsymbol{x}^{(k)} + \boldsymbol{f}. \tag{3.5}$$

对任意选取初始向量 $\boldsymbol{x}^{(0)}$，迭代法(3.5)收敛的充要条件是矩阵 $\boldsymbol{B}$ 的谱半径 $\rho(\boldsymbol{B})<1$.

证明 充分性. 设 $\rho(\boldsymbol{B})<1$，易知 $\boldsymbol{A}\boldsymbol{x}=\boldsymbol{f}$(其中 $\boldsymbol{A}=\boldsymbol{I}-\boldsymbol{B}$)有唯一解，记为 $\boldsymbol{x}^*$，则

$$\boldsymbol{x}^* = \boldsymbol{B}\boldsymbol{x}^* + \boldsymbol{f},$$

误差向量

$$\boldsymbol{\varepsilon}^{(k)} = \boldsymbol{x}^{(k)} - \boldsymbol{x}^* = \boldsymbol{B}^k\boldsymbol{\varepsilon}^{(0)},\ \boldsymbol{\varepsilon}^{(0)} = \boldsymbol{x}^{(0)} - \boldsymbol{x}^*.$$

由设 $\rho(\boldsymbol{B})<1$，应用定理 3，有$\lim\limits_{k\to\infty}\boldsymbol{B}^k=\boldsymbol{0}$. 于是对任意 $\boldsymbol{x}^{(0)}$ 有$\lim\limits_{k\to\infty}\boldsymbol{\varepsilon}^k$

$=\mathbf{0}$,即$\lim\limits_{k\to\infty}\boldsymbol{x}^{(k)}=\boldsymbol{x}^*$.

必要性. 设对任意 $\boldsymbol{x}^{(0)}$ 有

$$\lim_{k\to\infty}\boldsymbol{x}^{(k)} = \boldsymbol{x}^*,$$

其中 $\boldsymbol{x}^{(k+1)}=\boldsymbol{B}\boldsymbol{x}^{(k)}+\boldsymbol{f}$. 显然,极限 $\boldsymbol{x}^*$ 是方程组(3.4)的解,且对任意 $\boldsymbol{x}^{(0)}$ 有

$$\boldsymbol{\varepsilon}^{(k)} = \boldsymbol{x}^{(k)} - \boldsymbol{x}^* = \boldsymbol{B}^k\boldsymbol{\varepsilon}^{(0)} \to \mathbf{0} \quad (k\to\infty).$$

由定理 2 知

$$\lim_{k\to\infty}\boldsymbol{B}^k = \mathbf{0},$$

再由定理 3,即得 $\rho(\boldsymbol{B})<1$.

定理 4 是一阶定常迭代法的基本理论.

推论 设 $\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}$,其中 $\boldsymbol{A}=\boldsymbol{D}-\boldsymbol{L}-\boldsymbol{U}$ 为非奇异矩阵且 $\boldsymbol{D}$ 非奇异,则

(1) 解方程组的雅可比迭代法收敛的充要条件是 $\rho(\boldsymbol{J})<1$,其中 $\boldsymbol{J}=\boldsymbol{D}^{-1}(\boldsymbol{L}+\boldsymbol{U})$.

(2) 解方程组的高斯-塞德尔迭代法收敛的充要条件是 $\rho(\boldsymbol{G})<1$,其中 $\boldsymbol{G}=(\boldsymbol{D}-\boldsymbol{L})^{-1}\boldsymbol{U}$.

(3) 解方程组的 SOR 方法收敛的充要条件是 $\rho(\boldsymbol{L}_\omega)<1$,其中 $\boldsymbol{L}_\omega=(\boldsymbol{D}-\omega\boldsymbol{L})^{-1}((1-\omega)\boldsymbol{D}+\omega\boldsymbol{U})$.

例 5 考察用雅可比方法解方程组(1.2)的收敛性. 迭代矩阵 $\boldsymbol{J}$ 的特征方程为

$$\det(\lambda\boldsymbol{I}-\boldsymbol{J}) = \lambda^3 + 0.034090909\lambda + 0.039772727 = 0,$$

解得

$$\lambda_1 = -0.3082,$$
$$\lambda_2 = 0.1541 + \mathrm{i}0.3245,$$
$$\lambda_3 = 0.1541 - \mathrm{i}0.3245,$$
$$|\lambda_2| = |\lambda_3| = 0.3592 < 1,\ |\lambda_1| < 1,$$

即 $\rho(\boldsymbol{J})<1$. 所以用雅可比迭代法解方程组(1.2)是收敛的.

例 6　考察用迭代法解方程组

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{B}\boldsymbol{x}^{(k)} + \boldsymbol{f}$$

的收敛性,其中 $\boldsymbol{B}=\begin{pmatrix}0 & 2\\ 3 & 0\end{pmatrix}$, $\boldsymbol{f}=\begin{pmatrix}5\\ 5\end{pmatrix}$.

解　特征方程为 $\det(\lambda\boldsymbol{I}-\boldsymbol{B})=\lambda^2-6=0$,特征根 $\lambda_{1,2}=\pm\sqrt{6}$,即 $\rho(\boldsymbol{B})>1$. 这说明用迭代法解此方程组不收敛.

迭代法的基本定理在理论上是重要的,由于 $\rho(\boldsymbol{B})\leqslant\|\boldsymbol{B}\|$,下面利用矩阵 $\boldsymbol{B}$ 的范数建立判别迭代法收敛的充分条件.

定理 5(迭代法收敛的充分条件)　设有方程组

$$\boldsymbol{x} = \boldsymbol{B}\boldsymbol{x} + \boldsymbol{f},\ \boldsymbol{B} \in \mathbf{R}^{n\times n},$$

及一阶定常迭代法

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{B}\boldsymbol{x}^{(k)} + \boldsymbol{f}.$$

如果有 $\boldsymbol{B}$ 的某种算子范数 $\|\boldsymbol{B}\|=q<1$,则

(1) 迭代法收敛,即对任取 $\boldsymbol{x}^{(0)}$ 有

$$\lim_{k\to\infty}\boldsymbol{x}^{(k)} = \boldsymbol{x}^* \quad \text{且} \quad \boldsymbol{x}^* = \boldsymbol{B}\boldsymbol{x}^* + \boldsymbol{f}.$$

(2) $\|\boldsymbol{x}^*-\boldsymbol{x}^{(k)}\|\leqslant q^k\|\boldsymbol{x}^*-\boldsymbol{x}^{(0)}\|$.

(3) $\|\boldsymbol{x}^*-\boldsymbol{x}^{(k)}\|\leqslant\dfrac{q}{1-q}\|\boldsymbol{x}^{(k)}-\boldsymbol{x}^{(k-1)}\|$.

(4) $\|\boldsymbol{x}^*-\boldsymbol{x}^{(k)}\|\leqslant\dfrac{q^k}{1-q}\|\boldsymbol{x}^{(1)}-\boldsymbol{x}^{(0)}\|$.

证明　(1) 由基本定理 4 结论(1)是显然的.

(2) 显然有关系式 $\boldsymbol{x}^*-\boldsymbol{x}^{(k+1)}=\boldsymbol{B}(\boldsymbol{x}^*-\boldsymbol{x}^{(k)})$及

$$\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)} = \boldsymbol{B}(\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}).$$

于是有　(a) $\|\boldsymbol{x}^{(k+1)}-\boldsymbol{x}^{(k)}\|\leqslant q\|\boldsymbol{x}^{(k)}-\boldsymbol{x}^{(k-1)}\|$;

　　　　(b) $\|\boldsymbol{x}^*-\boldsymbol{x}^{(k+1)}\|\leqslant q\|x^*-x^{(k)}\|$.

反复利用(b)即得(2).

(3) 考查

$$\begin{aligned}\|x^{(k+1)}-x^{(k)}\| &= \|x^*-x^{(k)}-(x^*-x^{(k+1)})\| \\ &\geqslant \|x^*-x^{(k)}\|-\|x^*-x^{(k+1)}\| \\ &\geqslant (1-q)\|x^*-x^{(k)}\|,\end{aligned}$$

即
$$\begin{aligned}\|x^*-x^{(k)}\| &\leqslant \frac{1}{1-q}\|x^{(k+1)}-x^{(k)}\| \\ &\leqslant \frac{q}{1-q}\|x^{(k)}-x^{(k-1)}\|.\end{aligned}$$

(4) 反复利用(a)，则得到(4).

6.3.2 关于解某些特殊方程组迭代法的收敛性

在科学及工程计算中，要求解方程组 $\boldsymbol{Ax}=\boldsymbol{b}$，其矩阵 $\boldsymbol{A}$ 常常具有某些特性. 例如，$\boldsymbol{A}$ 具有对角占优性质或 $\boldsymbol{A}$ 为不可约阵，或 $\boldsymbol{A}$ 是对称正定阵等，下面讨论用基本迭代法解这些方程组的收敛性.

定义 3(对角占优阵) 设 $\boldsymbol{A}=(a_{ij})_{n\times n}$.

(1) 如果 $\boldsymbol{A}$ 的元素满足

$$|a_{ii}|>\sum_{\substack{j=1\\j\neq i}}^{n}|a_{ij}| \qquad (i=1,2,\cdots,n).$$

称 $\boldsymbol{A}$ 为**严格对角占优阵**.

(2) 如果 $\boldsymbol{A}$ 元素满足

$$|a_{ii}|\geqslant\sum_{\substack{j=1\\j\neq i}}^{n}|a_{ij}| \qquad (i=1,2,\cdots,n).$$

且上式至少有一个不等式严格成立，称 $\boldsymbol{A}$ 为**弱对角占优阵**.

定义 4(可约与不可约矩阵) 设 $\boldsymbol{A}=(a_{ij})_{n\times n}(n\geqslant 2)$，如果存在置换阵 $\boldsymbol{P}$ 使

$$\boldsymbol{P}^{\mathrm{T}}\boldsymbol{AP}=\begin{pmatrix}\boldsymbol{A}_{11} & \boldsymbol{A}_{12}\\ \boldsymbol{0} & \boldsymbol{A}_{22}\end{pmatrix}, \tag{3.6}$$

其中 $\boldsymbol{A}_{11}$ 为 r 阶方阵，$\boldsymbol{A}_{22}$ 为 $n-r$ 阶方阵($1\leqslant r<n$)，则称 $\boldsymbol{A}$ 为**可约**

矩阵,否则,如果不存在这样置换阵 $\boldsymbol{P}$ 使(3.6)式成立,则称 $\boldsymbol{A}$ 为**不可约矩阵**.

$\boldsymbol{A}$ 为可约矩阵意即 $\boldsymbol{A}$ 可经过若干行列重排化为(3.6)或 $\boldsymbol{Ax}=\boldsymbol{b}$ 可化为两个低阶方程组求解(如果 $\boldsymbol{A}$ 经过两行交换的同时进行相应两列的交换,称对 $\boldsymbol{A}$ 进行一次行列重排).

事实上,由 $\boldsymbol{Ax}=\boldsymbol{b}$ 可化为

$$\boldsymbol{P}^{\mathrm{T}}\boldsymbol{AP}(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{x}) = \boldsymbol{P}^{\mathrm{T}}\boldsymbol{b},$$

且记　$\boldsymbol{y}=\boldsymbol{P}^{\mathrm{T}}\boldsymbol{x}=\begin{pmatrix}\boldsymbol{y}_1\\\boldsymbol{y}_2\end{pmatrix}$, $\boldsymbol{P}^{\mathrm{T}}\boldsymbol{b}=\begin{pmatrix}\boldsymbol{d}_1\\\boldsymbol{d}_2\end{pmatrix}$,其中 $\boldsymbol{y}_1,\boldsymbol{d}_1$ 为 r 维向量. 于是,求解 $\boldsymbol{Ax}=\boldsymbol{b}$ 化为求解

$$\begin{cases}\boldsymbol{A}_{11}\boldsymbol{y}_1+\boldsymbol{A}_{12}\boldsymbol{y}_2=\boldsymbol{d}_1,\\ \qquad\qquad\ \boldsymbol{A}_{22}\boldsymbol{y}_2=\boldsymbol{d}_2.\end{cases}$$

由上式第 2 个方程组求出 $\boldsymbol{y}_2$,再代入第 1 个方程组求出 $\boldsymbol{y}_1$.

显然,如果 $\boldsymbol{A}$ 所有元素都非零,则 $\boldsymbol{A}$ 为不可约阵.

例 7　设有矩阵

$$\boldsymbol{A}=\begin{pmatrix}b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{n-1} & b_{n-1} & c_{n-1}\\ & & & a_n & b_n\end{pmatrix},\ \boldsymbol{B}=\begin{pmatrix}4 & -1 & -1 & 0\\ -1 & 4 & 0 & -1\\ -1 & 0 & 4 & -1\\ 0 & -1 & -1 & 4\end{pmatrix}.$$

$(a_i,b_i,c_i$ 都不为零)

则 $\boldsymbol{A},\boldsymbol{B}$ 都是不可约矩阵.

定理 6(对角占优定理)　如果 $\boldsymbol{A}=(a_{ij})_{n\times n}$ 为严格对角占优矩阵或 $\boldsymbol{A}$ 为不可约弱对角占优矩阵,则 $\boldsymbol{A}$ 为非奇异矩阵.

证明　只就 $\boldsymbol{A}$ 为严格对角占优阵证明此定理. 采用反证法,如果 $\det(\boldsymbol{A})=0$,则 $\boldsymbol{Ax}=\boldsymbol{0}$ 有非零解,记为 $\boldsymbol{x}=(x_1,\cdots,x_n)^{\mathrm{T}}$, 则 $|x_k|=\max\limits_{1\leqslant i\leqslant n}|x_i|\neq 0$.

由齐次方程组第 k 个方程

$$\sum_{j=1}^{n} a_{kj} x_j = 0,$$

则有

$$|a_{kk}x_k| = \left|\sum_{\substack{j=1\\ j\neq k}}^{n} a_{kj}x_j\right| \leqslant \sum_{\substack{j=1\\ j\neq k}}^{n} |a_{kj}||x_j| \leqslant |x_k| \sum_{\substack{j=1\\ j\neq k}}^{n} |a_{kj}|,$$

即
$$|a_{kk}| \leqslant \sum_{\substack{j=1\\ j\neq k}}^{n} |a_{kj}|,$$

与假设矛盾，故 $\det(\boldsymbol{A})\neq 0$.

定理 7 设 $\boldsymbol{Ax}=\boldsymbol{b}$，如果：

(1) $\boldsymbol{A}$ 为严格对角占优阵，则解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的雅可比迭代法，高斯-塞德尔迭代法均收敛.

(2) $\boldsymbol{A}$ 为弱对角占优阵，且 $\boldsymbol{A}$ 为不可约矩阵，则解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的雅可比迭代法，高斯-塞德尔迭代法均收敛.

证明 只证(1)中高斯-塞德尔迭代法收敛，其他同理可证.

由设可知，$a_{ii}\neq 0(i=1,\cdots,n)$，解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的高斯-塞德尔迭代法的迭代矩阵为 $\boldsymbol{G}=(\boldsymbol{D}-\boldsymbol{L})^{-1}\boldsymbol{U}$ $(\boldsymbol{A}=\boldsymbol{D}-\boldsymbol{L}-\boldsymbol{U})$. 下面考查 $\boldsymbol{G}$ 的特征值情况.

$$\begin{aligned}\det(\lambda\boldsymbol{I}-\boldsymbol{G}) &= \det(\lambda\boldsymbol{I}-(\boldsymbol{D}-\boldsymbol{L})^{-1}\boldsymbol{U})\\ &= \det((\boldsymbol{D}-\boldsymbol{L})^{-1})\cdot\det(\lambda(\boldsymbol{D}-\boldsymbol{L})-\boldsymbol{U}).\end{aligned}$$

由于 $\det((\boldsymbol{D}-\boldsymbol{L})^{-1})\neq 0$，于是 $\boldsymbol{G}$ 特征值即为 $\det(\lambda(\boldsymbol{D}-\boldsymbol{L})-\boldsymbol{U})=0$ 之根. 记

$$\boldsymbol{C}\equiv\lambda(\boldsymbol{D}-\boldsymbol{L})-\boldsymbol{U}=\begin{pmatrix}\lambda a_{11} & a_{12} & \cdots & a_{1n}\\ \lambda a_{21} & \lambda a_{22} & \cdots & a_{2n}\\ \vdots & \vdots & & \vdots\\ \lambda a_{n1} & \lambda a_{n2} & \cdots & \lambda a_{nn}\end{pmatrix},$$

下面来证明，当 $|\lambda|\geqslant 1$ 时，则 $\det(\boldsymbol{C})\neq 0$，即 $\boldsymbol{G}$ 的特征值均满足 $|\lambda|$

<1,由基本定理,则有高斯-塞德尔迭代法收敛.

事实上,当$|\lambda|\geqslant 1$时,由 $\boldsymbol{A}$ 为严格对角占优阵,则有

$$|c_{ii}|=|\lambda a_{ii}|>|\lambda|\left(\sum_{j=1}^{i-1}|a_{ij}|+\sum_{j=i+1}^{n}|a_{ij}|\right)$$

$$\geqslant\sum_{j=1}^{i-1}|\lambda a_{ij}|+\sum_{j=i+1}^{n}|a_{ij}|=\sum_{\substack{j=1\\j\neq i}}^{n}|c_{ij}|$$

$$(i=1,2,\cdots,n).$$

这说明,当$|\lambda|\geqslant 1$时,矩阵 $\boldsymbol{C}$ 为严格对角占优阵,再由对角占优定理有 $\det(\boldsymbol{C})\neq 0$.

下面研究对于解方程组 $\boldsymbol{Ax}=\boldsymbol{b}$ 的 SOR 方法中松弛因子 ω 在什么范围内取值,SOR 方法才可能收敛.

定理 8(SOR 方法收敛的必要条件) 设解方程组 $\boldsymbol{Ax}=\boldsymbol{b}$ 的 SOR 迭代法收敛,则 $0<\omega<2$.

证明 由设 SOR 迭代法收敛,则由定理 4 的推论中的(3)有 $\rho(\boldsymbol{L}_\omega)<1$,设 $\boldsymbol{L}_\omega$ 的特征值为 $\lambda_1,\lambda_2,\cdots,\lambda_n$,则

$$|\det(\boldsymbol{L}_\omega)|=|\lambda_1\lambda_2\cdots\lambda_n|\leqslant[\rho(\boldsymbol{L}_\omega)]^n,$$

或

$$|\det(\boldsymbol{L}_\omega)|^{1/n}\leqslant\rho(\boldsymbol{L}_\omega)<1.$$

另一方面

$$\begin{aligned}\det(\boldsymbol{L}_\omega)&=\det[(\boldsymbol{D}-\omega\boldsymbol{L})^{-1}]\det((1-\omega)\boldsymbol{D}+\omega\boldsymbol{U})\\&=(1-\omega)^n,\end{aligned}$$

从而

$$|\det(\boldsymbol{L}_\omega)|^{1/n}=|1-\omega|<1,$$

即

$$0<\omega<2.$$

定理 8 说明解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的 SOR 迭代法,只有在(0,2)范围内取松弛因子 ω,才可能收敛.

定理 9 设 $\boldsymbol{Ax}=\boldsymbol{b}$,如果:

(1) $\boldsymbol{A}$ 为对称正定矩阵,$\boldsymbol{A}=\boldsymbol{D}-\boldsymbol{L}-\boldsymbol{U}$;

(2) $0<\omega<2$.

则解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的 SOR 迭代法收敛.

证明 在上述假定下,若能证明 $|\lambda|<1$,那么定理得证(其中 λ 为 $\boldsymbol{L}_\omega$ 的任一特征值).

事实上,设 $\boldsymbol{y}$ 为对应 λ 的 $\boldsymbol{L}_\omega$ 的特征向量,即

$$\boldsymbol{L}_\omega \boldsymbol{y} = \lambda \boldsymbol{y},\ \boldsymbol{y} = (y_1, y_2, \cdots, y_n)^{\mathrm{T}} \neq \boldsymbol{0},$$

$$(\boldsymbol{D} - \omega\boldsymbol{L})^{-1}((1-\omega)\boldsymbol{D} + \omega\boldsymbol{U})\boldsymbol{y} = \lambda\boldsymbol{y},$$

亦即

$$((1-\omega)\boldsymbol{D} + \omega\boldsymbol{U})\boldsymbol{y} = \lambda(\boldsymbol{D} - \omega\boldsymbol{L})\boldsymbol{y}.$$

为了找出 λ 的表达式,考虑数量积

$$(((1-\omega)\boldsymbol{D} + \omega\boldsymbol{U})\boldsymbol{y}, \boldsymbol{y}) = \lambda((\boldsymbol{D} - \omega\boldsymbol{L})\boldsymbol{y}, \boldsymbol{y}),$$

则

$$\lambda = \frac{(\boldsymbol{Dy}, \boldsymbol{y}) - \omega(\boldsymbol{Dy}, \boldsymbol{y}) + \omega(\boldsymbol{Uy}, \boldsymbol{y})}{(\boldsymbol{Dy}, \boldsymbol{y}) - \omega(\boldsymbol{Ly}, \boldsymbol{y})},$$

显然

$$(\boldsymbol{Dy}, \boldsymbol{y}) = \sum_{i=1}^{n} a_{ii} \mid y_i \mid^2 \equiv \sigma > 0, \tag{3.7}$$

记

$$-(\boldsymbol{Ly}, \boldsymbol{y}) = \alpha + \mathrm{i}\beta,$$

由于 $\boldsymbol{A}=\boldsymbol{A}^{\mathrm{T}}$,所以 $\boldsymbol{U}=\boldsymbol{L}^{\mathrm{T}}$,故

$$-(\boldsymbol{Uy}, \boldsymbol{y}) = -(\boldsymbol{y}, \boldsymbol{Ly}) = -(\overline{\boldsymbol{Ly}, \boldsymbol{y}}) = \alpha - \mathrm{i}\beta,$$

$$0 < (\boldsymbol{Ay}, \boldsymbol{y}) = ((\boldsymbol{D} - \boldsymbol{L} - \boldsymbol{U})\boldsymbol{y}, \boldsymbol{y}) = \sigma + 2\alpha, \tag{3.8}$$

所以

$$\lambda = \frac{(\sigma - \omega\sigma - \alpha\omega) + \mathrm{i}\omega\beta}{(\sigma + \alpha\omega) + \mathrm{i}\omega\beta},$$

从而

$$|\lambda|^2 = \frac{(\sigma - \omega\sigma - \alpha\omega)^2 + \omega^2\beta^2}{(\sigma + \alpha\omega)^2 + \omega^2\beta^2}.$$

当 $0<\omega<2$ 时,利用(3.7),(3.8),有

$$(\sigma-\omega\sigma-\alpha\omega)^2-(\sigma+\alpha\omega)^2=\omega\sigma(\sigma+2\sigma)(\omega-2)<0,$$

即 $\boldsymbol{L}_\omega$ 的任一特征值满足 $|\lambda|<1$,故 SOR 方法收敛(注意当 $0<\omega<2$ 时,可以证明 $(\sigma+2\omega)^2+\omega^2\beta^2\neq0$).

定理 10　设 $\boldsymbol{Ax}=\boldsymbol{b}$,如果:

(1) $\boldsymbol{A}$ 为严格对角占优矩阵(或 $\boldsymbol{A}$ 为弱对角占优不可约矩阵);

(2) $0<\omega\leqslant1$.

则解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的 SOR 迭代法收敛.

下面讨论迭代法的收敛速度.

由定理 3 证明中可知,如果 $\rho(\boldsymbol{B})<1$ 且 $\rho(\boldsymbol{B})$ 越小时,迭代法收敛越快. 现设有方程组

$$\boldsymbol{x}=\boldsymbol{Bx}+\boldsymbol{f},\ \boldsymbol{B}\in\mathbf{R}^{n\times n}$$

及一阶定常迭代法

$$\boldsymbol{x}^{(k+1)}=\boldsymbol{Bx}^{(k)}+\boldsymbol{f}\quad(k=0,1,\cdots),\tag{3.9}$$

且设迭代法收敛,记 $\lim\limits_{k\to\infty}\boldsymbol{x}^{(k)}=\boldsymbol{x}^*$,则 $\boldsymbol{x}^*=\boldsymbol{Bx}^*+\boldsymbol{f}$.

由基本定理有 $0<\rho(\boldsymbol{B})<1$, 且误差向量 $\boldsymbol{\varepsilon}^{(k)}=\boldsymbol{x}^{(k)}-\boldsymbol{x}^*$ 满足

$$\boldsymbol{\varepsilon}^{(k)}=\boldsymbol{B}^k\boldsymbol{\varepsilon}^{(0)},$$

故

$$\|\boldsymbol{\varepsilon}^{(k)}\|\leqslant\|\boldsymbol{B}\|^k\|\boldsymbol{\varepsilon}^{(0)}\|.$$

现设 $\boldsymbol{B}$ 为对称矩阵,则有

$$\begin{aligned}\|\boldsymbol{\varepsilon}^{(k)}\|_2&\leqslant\|\boldsymbol{B}\|_2^k\|\boldsymbol{\varepsilon}^{(0)}\|_2\\&=[\rho(\boldsymbol{B})]^k\|\boldsymbol{\varepsilon}^{(0)}\|_2.\end{aligned}$$

下面确定欲使初始误差缩小 10^{-s} 所需的迭代次数,即欲使

$$[\rho(\boldsymbol{B})]^k\leqslant10^{-s}.$$

取对数,得到所需最少迭代次数为

$$k\geqslant\frac{s\ln10}{-\ln\rho(\boldsymbol{B})}.\tag{3.10}$$

这说明,所需迭代次数与 $R=-\ln\rho(\boldsymbol{B})$ 成反比,当 $\rho(\boldsymbol{B})<1$ 越小,

$R=-\ln\rho(\boldsymbol{B})$越大，则(3.10)式满足所需迭代次数越少，即迭代法收敛越快. 当迭代矩阵 $\boldsymbol{B}$ 为一般矩阵时的讨论可参考[7].

定义 5 称 $R(\boldsymbol{B})=-\ln\rho(\boldsymbol{B})$ 为迭代法(3.9)的**渐近收敛速度**，简称**迭代法收敛速度**.

对于 SOR 迭代法希望选择松弛因子 ω 使迭代过程(2.10)收敛较快，在理论上即确定 ω_{opt} 使

$$\min_{0<\omega<2}\rho(\boldsymbol{L}_\omega)=\rho(\boldsymbol{L}_{\omega_{\text{opt}}}).$$

对某些特殊类型的矩阵，建立了 SOR 方法最佳松弛因子理论. 例如，对所谓具有"性质 A"等条件的线性方程组建立了最佳松弛因子公式

$$\omega_{\text{opt}}=\frac{2}{1+\sqrt{1-(\rho(\boldsymbol{J}))^2}},$$

其中 $\rho(\boldsymbol{J})$ 为解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的雅可比迭代法的迭代矩阵的谱半径.

在实际应用中，对于某些椭圆型微分方程(模型问题)可以给出 ω_{opt} 的计算方法，但一般来说，计算 ω_{opt} 是有困难的，可用试算的办法来确定一个适当的 ω.

算法 2(SOR 迭代法) 设 $\boldsymbol{Ax}=\boldsymbol{b}$，其中 $\boldsymbol{A}$ 为对称正定矩阵或为严格对角占优阵或为弱对角占优不可约矩阵等，本算法用 SOR 迭代法求解 $\boldsymbol{Ax}=\boldsymbol{b}$，数组 $x(n)$ 存放 $\boldsymbol{x}^{(0)}$ 及 $\boldsymbol{x}^{(k)}$，用 $p_0\max\limits_{1\leq i\leq n}|\Delta x_i|<$ eps 控制迭代终止，用 N_0 表示最大迭代次数.

1. $k\leftarrow 0$
2. $x_i\leftarrow 0.0\quad(i=1,2,\cdots,n)$
3. $k\leftarrow k+1$
4. $p_0\leftarrow 0.0$
5. 对于 $i=1,2,\cdots,n$

(1) $p\leftarrow\Delta x_i=\omega*\left(b_i-\sum\limits_{j=1}^{i-1}a_{ij}x_j-\sum\limits_{j=i}^{n}a_{ij}x_j\right)\Big/a_{ii}$

(2) 如果 $|p|>p_0$ 则 $p_0 \leftarrow |p|$

(3) $x_i \leftarrow x_i + p$

6. 输出 p_0

7. 如果 $p_0 < eps$ 则输出 $k,\omega,\boldsymbol{x}$,停机

8. 如果 $k<N_0$ 则转 3

9. 输出 N_0 及有关信息

[注] 可用 $\| \boldsymbol{r}^{(k)} \|_\infty < eps$ 来控制迭代终止,其中 $\boldsymbol{r}^{(k)} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{(k)}$.

6.4 分块迭代法

上述迭代法,从 $\boldsymbol{x}^{(k)} \to \boldsymbol{x}^{(k+1)}$ 计算时,是逐个计算 $\boldsymbol{x}^{(k+1)}$ 的分量 $x_i^{(k+1)}(i=1,2,\cdots,n)$,这种迭代法又称为点迭代法,下面介绍分块迭代法,就是一块或一组未知数同时被改进.

设 $\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}$,其中 $\boldsymbol{A} \in \mathbf{R}^{n\times n}$ 为大型稀疏矩阵且将 $\boldsymbol{A}$ 分块为三部分 $\boldsymbol{A}=\boldsymbol{D}-\boldsymbol{L}-\boldsymbol{U}$,其中

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{A}_{11} & \boldsymbol{A}_{12} & \cdots & \boldsymbol{A}_{1q} \\ \boldsymbol{A}_{21} & \boldsymbol{A}_{22} & \cdots & \boldsymbol{A}_{2q} \\ \vdots & \vdots & & \vdots \\ \boldsymbol{A}_{q1} & \boldsymbol{A}_{q2} & \cdots & \boldsymbol{A}_{qq} \end{pmatrix}, \quad \boldsymbol{D} = \begin{pmatrix} \boldsymbol{A}_{11} & & & \\ & \boldsymbol{A}_{22} & & \\ & & \ddots & \\ & & & \boldsymbol{A}_{qq} \end{pmatrix},$$

$$\boldsymbol{L} = \begin{pmatrix} \boldsymbol{0} & & & \\ -\boldsymbol{A}_{21} & \boldsymbol{0} & & \\ \vdots & \vdots & \ddots & \\ -\boldsymbol{A}_{q1} & -\boldsymbol{A}_{q2} & \cdots & \boldsymbol{0} \end{pmatrix}, \quad \boldsymbol{U} = \begin{pmatrix} \boldsymbol{0} & -\boldsymbol{A}_{12} & \cdots & -\boldsymbol{A}_{1q} \\ & \boldsymbol{0} & \cdots & -\boldsymbol{A}_{2q} \\ & & \ddots & \vdots \\ & & & \boldsymbol{0} \end{pmatrix}.$$

且 $\boldsymbol{A}_{ii}(i=1,2,\cdots,q)$ 为 $n_i \times n_i$ 非奇异矩阵, $\sum_{i=1}^{q} n_i = n$. 对 $\boldsymbol{x}$ 及 $\boldsymbol{b}$ 同样分块

$$\boldsymbol{x}=\begin{pmatrix}\boldsymbol{x}_1\\\boldsymbol{x}_2\\\vdots\\\boldsymbol{x}_q\end{pmatrix},\ \boldsymbol{b}=\begin{pmatrix}\boldsymbol{b}_1\\\boldsymbol{b}_2\\\vdots\\\boldsymbol{b}_q\end{pmatrix},$$

其中,$\boldsymbol{x}_i\in\mathbf{R}^{n_i}$, $\boldsymbol{b}_i\in\mathbf{R}^{n_i}$.

(1) 块雅可比迭代法(BJ)

选取分裂阵 $\boldsymbol{M}$ 为 $\boldsymbol{A}$ 的对角块部分,即选

$$\begin{cases}\boldsymbol{M}=\boldsymbol{D}(\text{块对角阵}),\\\boldsymbol{A}=\boldsymbol{M}-\boldsymbol{N}.\end{cases}$$

于是,得到块雅可比迭代法

$$\boldsymbol{x}^{(k+1)}=\boldsymbol{B}\boldsymbol{x}^{(k)}+\boldsymbol{f},\tag{4.1}$$

其中迭代矩阵 $\boldsymbol{B}=\boldsymbol{I}-\boldsymbol{D}^{-1}\boldsymbol{A}=\boldsymbol{D}^{-1}(\boldsymbol{L}+\boldsymbol{U})\equiv\boldsymbol{J}$, $\boldsymbol{f}=\boldsymbol{D}^{-1}\boldsymbol{b}$.
或

$$\boldsymbol{D}\boldsymbol{x}^{(k+1)}=(\boldsymbol{L}+\boldsymbol{U})\boldsymbol{x}^{(k)}+\boldsymbol{b}.$$

由分块矩阵乘法,得到块雅可比迭代法的具体形式

$$\boldsymbol{A}_{ii}\boldsymbol{x}_i^{(k+1)}=\boldsymbol{b}_i-\sum_{\substack{j=1\\j\neq i}}^{q}\boldsymbol{A}_{ij}\boldsymbol{x}_j^{(k)}.\quad(i=1,2,\cdots,q),\tag{4.2}$$

其中

$$\boldsymbol{x}^{(k)}=\begin{pmatrix}\boldsymbol{x}_1^{(k)}\\\boldsymbol{x}_2^{(k)}\\\vdots\\\boldsymbol{x}_q^{(k)}\end{pmatrix},\ \boldsymbol{x}_i^{(k)}\in\mathbf{R}^{n_i}.$$

这说明,块雅可比迭代法,每迭代一步,从 $\boldsymbol{x}^{(k)}\to\boldsymbol{x}^{(k+1)}$,需要求解 q 个低阶方程组

$$\begin{cases}\boldsymbol{A}_{ii}\boldsymbol{x}_i^{(k+1)}=\boldsymbol{g}_i\\(i=1,2,\cdots,q),\text{其中 }\boldsymbol{g}_i\text{ 为(3.12)式右边部分.}\end{cases}$$

(2) 块 SOR 迭代法(BSOR)

选取分裂矩阵 $\boldsymbol{M}$ 为带松弛因子的 $\boldsymbol{A}$ 块下三角部分，即

$$\begin{cases}\boldsymbol{M} = \dfrac{1}{\omega}(\boldsymbol{D} - \omega\boldsymbol{L}),\\ \boldsymbol{A} = \boldsymbol{M} - \boldsymbol{N}.\end{cases}$$

得到块 SOR 迭代法

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{L}_\omega \boldsymbol{x}^{(k)} + \boldsymbol{f}, \tag{4.3}$$

其中迭代矩阵

$$\begin{aligned}\boldsymbol{L}_\omega &= \boldsymbol{I} - \omega(\boldsymbol{D} - \omega\boldsymbol{L})^{-1}\boldsymbol{A}\\ &= (\boldsymbol{D} - \omega\boldsymbol{L})^{-1}((1-\omega)\boldsymbol{D} + \omega\boldsymbol{U}),\\ \boldsymbol{f} &= \omega(\boldsymbol{D} - \omega\boldsymbol{L})^{-1}\boldsymbol{b}.\end{aligned}$$

由分块矩阵乘法得到块 SOR 迭代法的具体形式

$$\begin{cases}\boldsymbol{A}_{ii}\boldsymbol{x}_i^{(k+1)} = \boldsymbol{A}_{ii}\boldsymbol{x}_i^{(k)} + \omega\left(\boldsymbol{b}_i - \sum\limits_{j=1}^{i-1}\boldsymbol{A}_{ij}\boldsymbol{x}_j^{(k+1)} - \sum\limits_{j=i}^{q}\boldsymbol{A}_{ij}\boldsymbol{x}_j^{(k)}\right)\\ (i = 1,2,\cdots,q;\ k = 0,1,\cdots),\\ \omega \text{ 为松弛因子}.\end{cases} \tag{4.4}$$

于是，当 $\boldsymbol{x}^{(k)}$ 及 $\boldsymbol{x}_j^{(k+1)}(j=1,2,\cdots,i-1)$ 已计算时，解低阶方程组(3.14)可计算小块 $\boldsymbol{x}_i^{(k+1)}$. 从 $\boldsymbol{x}^{(k)} \to \boldsymbol{x}^{(k+1)}$ 共需要解 q 个低阶方程组，当 $\boldsymbol{A}_{ii}$ 为三对角阵或带状矩阵时，可用直接法求解.

我们给出下述结果.

定理 11　设 $\boldsymbol{Ax}=\boldsymbol{b}$，其中 $\boldsymbol{A}=\boldsymbol{D}-\boldsymbol{L}-\boldsymbol{U}$(分块形式).

(1) 如果 $\boldsymbol{A}$ 为对称正定矩阵，

(2) $0<\omega<2$.

则解 $\boldsymbol{Ax}=\boldsymbol{b}$ 的 BSOR 迭代法收敛.

评　注

本章介绍了解大型稀疏线性方程组的一些基本迭代法，例如，雅可比迭代法，高斯-塞德尔迭代法，及 SOR 迭代法，分块迭代法等. 并且建立了迭代法的一些基本理论. 在应用中 SOR 方法较为重要，它是解大型、稀疏线性方程组的有效方法之一.

迭代法是一种逐次逼近方法，在使用迭代法解方程组时，其系数矩阵在计算过程中始终不变.

迭代法具有循环的计算公式，方法简单，适宜解大型稀疏矩阵方程组，在计算机实现时只需存储 $\boldsymbol{A}$ 的非零元素(或可按一定公式形成系数，这样 $\boldsymbol{A}$ 就不需存储).

在使用迭代法时，要注意收敛性及收敛速度问题，使用 SOR 方法要选择较佳松弛因子.

迭代法的进一步学习，读者可参考文献[3]，[14]，[15]，[16].

习　题

1. 设方程组

$$\begin{cases} 5x_1+2x_2+x_3=-12, \\ -x_1+4x_2+2x_3=20, \\ 2x_1-3x_2+10x_3=3. \end{cases}$$

(a) 考察用雅可比迭代法，高斯-塞德尔迭代法解此方程组的收敛性；

(b) 用雅可比迭代法及高斯-塞德尔迭代法解此方程组，要求当 $\|\boldsymbol{x}^{(k+1)}-\boldsymbol{x}^{(k)}\|_\infty<10^{-4}$ 时迭代终止.

2. 设方程组

(a) $\begin{cases} x_1+0.4x_2+0.4x_3=1, \\ 0.4x_1+x_2+0.8x_2=2, \\ 0.4x_1+0.8x_2+x_3=3. \end{cases}$　　(b) $\begin{cases} x_1+2x_2-2x_3=1, \\ x_1+x_2+x_3=1, \\ 2x_1+2x_2+x_3=1. \end{cases}$

试考察解此方程组的雅可比迭代法及高斯-塞德尔迭代法的收敛性.

3. 求证$\lim\limits_{k\to\infty}\boldsymbol{A}_k=\boldsymbol{A}$的充要条件是对任何向量$\boldsymbol{x}$都有

$$\lim_{k\to\infty}\boldsymbol{A}_k\boldsymbol{x}=\boldsymbol{A}\boldsymbol{x}.$$

4. 设$\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}$,其中$\boldsymbol{A}$对称正定,问解此方程组的雅可比迭代法是否一定收敛?试考察习题 2(a)方程组.

5. 用 SOR 方法解方程组(分别取松弛因子$\omega=1.03$, $\omega=1$, $\omega=1.1$)

$$\begin{cases} 4x_1-x_2=1,\\ -x_1+4x_2-x_3=4,\\ -x_2+4x_3=-3. \end{cases}$$

精确解$\boldsymbol{x}^*=\left(\frac{1}{2},1,-\frac{1}{2}\right)^{\mathrm{T}}$. 要求当$\|\boldsymbol{x}^*-\boldsymbol{x}^{(k)}\|_\infty<5\times10^{-6}$时迭代终止,并且对每一个$\omega$值确定迭代次数.

6. 用 SOR 方法解方程组(取$\omega=0.9$)

$$\begin{cases} 5x_1+2x_2+x_3=-12,\\ -x_1+4x_2+2x_3=20,\\ 2x_1-3x_2+10x_3=3. \end{cases}$$

要求当$\|\boldsymbol{x}^{(k+1)}-\boldsymbol{x}^{(k)}\|_\infty<10^{-4}$时迭代终止.

7. 设有方程组$\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}$,其中$\boldsymbol{A}$为对称正定阵,迭代公式

$$\boldsymbol{x}^{(k+1)}=\boldsymbol{x}^{(k)}+\omega(\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x}^{(k)})\quad(k=0,1,2,\cdots),$$

试证明当$0<\omega<\frac{2}{\beta}$时上述迭代法收敛(其中$0<\alpha\leqslant\lambda(\boldsymbol{A})\leqslant\beta$).

8. 证明矩阵

$$\boldsymbol{A}=\begin{pmatrix}1&a&a\\a&1&a\\a&a&1\end{pmatrix}$$

对于$-\frac{1}{2}<a<1$是正定的,而雅可比迭代只对$-\frac{1}{2}<a<\frac{1}{2}$是收敛的.

9. 给定迭代过程$\boldsymbol{x}^{(k+1)}=\boldsymbol{G}\boldsymbol{x}^{(k)}+\boldsymbol{g}$, 其中$\boldsymbol{G}\in\mathbf{R}^{n\times n}$ $(k=0,1,2,\cdots)$,试证明:如果$\boldsymbol{G}$的特征值$\lambda_i(\boldsymbol{G})=0$ $(i=1,2,\cdots,n)$,则此迭代过程最多迭代n次收敛于方程组的解.

10. 设$\boldsymbol{A}$为严格对角占优阵,且$0<\omega\leqslant1$. 求证解$\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b}$的 SOR 迭代法收敛.

第7章　非线性方程求根

7.1　方程求根与二分法

7.1.1　引言

本章主要讨论单变量非线性方程

$$f(x)=0 \tag{1.1}$$

的求根问题，这里 $x\in\mathbf{R}$，$f(x)\in C[a,b]$. 在科学与工程计算中有大量方程求根问题，其中一类特殊的问题是多项式方程

$$f(x)=a_0x^n+a_1x^{n-1}+\cdots+a_{n-1}x+a_n \quad (a_0\neq 0), \tag{1.2}$$

其中系数 $a_i(i=0,1,\cdots,n)$ 为实数.

方程 $f(x)=0$ 的根 x^*，又称为函数 $f(x)$ 的零点，它使 $f(x^*)=0$；若 $f(x)$ 可分解为

$$f(x)=(x-x^*)^m g(x),$$

其中 m 为正整数，且 $g(x^*)\neq 0$. 当 $m=1$ 时，则称 x^* 为单根，若 $m>1$ 称 x^* 为(1.1)的 m 重根，或 x^* 为 $f(x)$ 的 m 重零点. 若 x^* 是 $f(x)$ 的 m 重零点，且 $g(x)$ 充分光滑，则

$$f(x^*)=f'(x^*)=\cdots=f^{(m-1)}(x^*)=0,\ f^{(m)}(x^*)\neq 0.$$

当 $f(x)$ 为代数多项式(1.2)时，根据代数基本定理可知，n 次方程在复数域有且只有 n 个根(含复根，m 重根为 m 个根)，$n=1$，2 时方程的根是大家熟悉的，$n=3,4$ 时虽有求根公式但比较复杂，可在数学手册中查到，但已不适合于数值计算，而 $n\geqslant 5$ 时就不能用公式表示方程的根. 因此，通常对 $n\geqslant 3$ 的多项式方程求根与一般连续函数方程(1.1)一样都可采用迭代法求根. 迭代法要求先

给出根 x^* 的一个近似,若 $f(x)\in C[a,b]$ 且 $f(a)f(b)<0$,根据连续函数性质可知 $f(x)=0$ 在 (a,b) 内至少有一个实根,这时称 $[a,b]$ 为方程(1.1)的有根区间.通常可通过逐次搜索法求得方程(1.1)的有根区间.

例1 求方程 $f(x)=x^3-11.1x^2+38.8x-41.77=0$ 的有根区间.

解 根据有根区间定义,对 $f(x)=0$ 的根进行搜索计算,结果如下:

x	0	1	2	3	4	5	6
$f(x)$的符号	−	−	+	+	−	−	+

由此可知方程的有根区间为[1,2],[3,4],[5,6].

7.1.2 二分法

考察有根区间 $[a,b]$,取中点 $x_0=(a+b)/2$ 将它分为两半,假设中点 x_0 不是 $f(x)$ 的零点,然后进行根的搜索,即检查 $f(x_0)$ 与 $f(a)$ 是否同号,如果确系同号,说明所求的根 x^* 在 x_0 的右侧,这时令 $a_1=x_0,b_1=b$;否则 x^* 必在 x_0 的左侧,这时令 $a_1=a,b_1=x_0$(图7-1).不管出现哪一种情况,新的有根区间 $[a_1,b_1]$ 的长度仅为 $[a,b]$ 的一半.

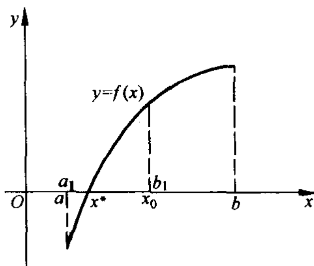


图 7-1

对压缩了的有根区间 $[a_1,b_1]$ 又可施行同样的手续,即用中点 $x_1=(a_1+b_1)/2$ 将区间 $[a_1,b_1]$ 再分为两半,然后通过根的搜索判定所求的根在 x_1 的哪一侧,从而又确定一个新的有根区间 $[a_2,b_2]$,其长度是 $[a_1,b_1]$ 的一半.

如此反复二分下去，即可得出一系列有根区间

$$[a,b] \supset [a_1,b_1] \supset [a_2,b_2] \supset \cdots \supset [a_k,b_k] \supset \cdots,$$

其中每个区间都是前一个区间的一半，因此$[a_k,b_k]$的长度

$$b_k - a_k = (b-a)/2^k$$

当 $k\to\infty$ 时趋于零，就是说，如果二分过程无限地继续下去，这些区间最终必收缩于一点 x^*，该点显然就是所求的根.

每次二分后，设取有根区间$[a_k,b_k]$的中点

$$x_k = (a_k + b_k)/2$$

作为根的近似值，则在二分过程中可以获得一个近似根的序列

$$x_0, x_1, x_2, \cdots, x_k, \cdots,$$

该序列必以根 x^* 为极限.

不过在实际计算时，我们不可能完成这个无限过程，其实也没有这种必要，因为数值分析的结果允许带有一定的误差. 由于

$$|x^* - x_k| \leqslant (b_k - a_k)/2 = (b-a)/2^{k+1}, \qquad (1.3)$$

只要二分足够多次（即 k 充分大），便有

$$|x^* - x_k| < \varepsilon,$$

这里 ε 为预定的精度.

例 2 求方程

$$f(x) = x^3 - x - 1 = 0$$

在区间[1.0,1.5]内的一个实根，要求准确到小数点后的第 2 位.

解 这里 $a=1.0$，$b=1.5$，而 $f(a)<0$，$f(b)>0$. 取$[a,b]$的中点 $x_0=1.25$，将区间二等分，由于 $f(x_0)<0$，即 $f(x_0)$与 $f(a)$同号，故所求的根 x^* 必在 x_0 右侧，这时应令 $a_1=x_0=1.25$，$b_1=b=1.5$，而得到新的有根区间$[a_1,b_1]$.

如此反复二分下去. 二分过程无需赘述. 我们现在预估所要二分的次数，按误差估计(1.3)式，只要二分 6 次（$k=6$），便能达到预定的精度

$$|x^* - x_6| \leqslant 0.005.$$

二分法的计算结果如表 7-1.

二分法是计算机上的一种常用算法，下面列出计算步骤：

表 7-1 计算结果

k	a_k	b_k	x_k	$f(x_k)$符号
0	1.0	1.5	1.25	−
1	1.25		1.375	+
2		1.375	1.3125	−
3	1.3125		1.3438	+
4		1.3438	1.3281	+
5		1.3281	1.3203	−
6	1.3203		1.3242	−

步骤 1 准备 计算 $f(x)$在有根区间$[a,b]$端点处的值 $f(a)$，$f(b)$.

步骤 2 二分 计算 $f(x)$在区间中点$\frac{a+b}{2}$处的值 $f\left(\frac{a+b}{2}\right)$.

步骤 3 判断 若 $f\left(\frac{a+b}{2}\right)=0$，则$\frac{a+b}{2}$即是根，计算过程结束，否则检验.

若 $f\left(\frac{a+b}{2}\right)f(a)<0$，则以$\frac{a+b}{2}$代替 b，否则以$\frac{a+b}{2}$代替 a.

反复执行步骤 2 和步骤 3，直到区间$[a,b]$长度小于允许误差 ε，此时中点$\frac{a+b}{2}$即为所求近似根.

上述二分法的优点是算法简单，且总是收敛的，缺点是收敛太慢，故一般不单独将其用于求根，只用其为根求得一个较好的近似值.

7.2 迭代法及其收敛性

7.2.1 不动点迭代法

将方程(1.1)改写成等价的形式

$$x = \varphi(x). \tag{2.1}$$

若要求 x^* 满足 $f(x^*)=0$,则 $x^*=\varphi(x^*)$;反之亦然,称 x^* 为函数 $\varphi(x)$的一个**不动点**. 求 $f(x)$的零点就等价于求 $\varphi(x)$的不动点,选择一个初始近似值 x_0,将它代入(2.1)右端,即可求得

$$x_1 = \varphi(x_0).$$

可以如此反复迭代计算

$$x_{k+1} = \varphi(x_k) \qquad (k = 0,1,\cdots). \tag{2.2}$$

$\varphi(x)$称为迭代函数. 如果对任何 $x_0 \in [a,b]$,由(2.2)得到的序列 $\{x_k\}$有极限

$$\lim_{k\to\infty} x_k = x^*.$$

则称迭代方程(2.2)收敛,且 $x^*=\varphi(x^*)$为 $\varphi(x)$的不动点,故称(2.2)为**不动点迭代法**.

上述迭代法是一种逐次逼近法,其基本思想是将隐式方程(2.1)归结为一组显式的计算公式(2.2). 就是说,迭代过程实质上是一个逐步显示化的过程.

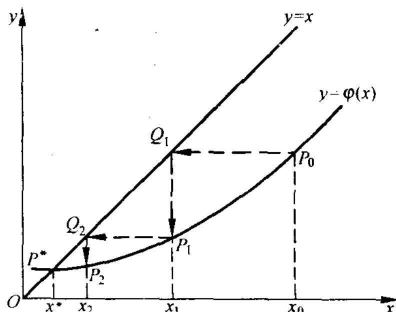


图 7-2

我们用几何图像来显示迭代过程. 方程 $x=\varphi(x)$的求根问题在 xy 平面上就是要确定曲线 $y=\varphi(x)$与直线 $y=x$ 的交点 P^*(图 7-2). 对于 x^*

的某个近似值 x_0，在曲线 $y=\varphi(x)$ 上可确定一点 P_0，它以 x_0 为横坐标，而纵坐标则等于 $\varphi(x_0)=x_1$. 过 P_0 引平行 x 轴的直线，设此直线交直线 $y=x$ 于点 Q_1，然后过 Q_1 再作平行于 y 轴的直线，它与曲线 $y=\varphi(x)$ 的交点记作 P_1，则点 P_1 的横坐标为 x_1，纵坐标则等于 $\varphi(x_1)=x_2$. 按图 7-2 中箭头所示的路径继续做下去，在曲线 $y=\varphi(x)$ 上得到点列 $P_1,P_2,\cdots$，其横坐标分别为依公式 $x_{k+1}=\varphi(x_k)$ 求得的迭代值 $x_1,x_2,\cdots$. 如果点列 $\{P_k\}$ 趋向于点 P^*，则相应的迭代值 x_k 收敛到所求的根 x^*.

例 3 求方程

$$f(x)=x^3-x-1=0 \tag{2.3}$$

在 $x_0=1.5$ 附近的根 x^*.

解 设将方程(2.3)改写成下列形式

$$x=\sqrt[3]{x+1}.$$

据此建立迭代公式

$$x_{k+1}=\sqrt[3]{x_k+1}\quad (k=0,1,2,\cdots).$$

表 7-2 记录了各步迭代的结果. 我们看到，如果仅取 6 位数字，那么结果 x_7 与 x_8 完全相同，这时可以认为 x_7 实际上已满足方程(2.3)，即为所求的根.

表 7-2 计算结果

k	x_k	k	x_k
0	1.5	5	1.32476
1	1.35721	6	1.32473
2	1.33086	7	1.32472
3	1.32588	8	1.32472
4	1.32494		

应当指出，迭代法的效果并不是总能令人满意的. 譬如，设依方程(2.3)的另一种等价形式

$$x=x^3-1$$

建立迭代公式

$$x_{k+1}=x_k^3-1.$$

迭代初值仍取 $x_0=1.5$，则有

$$x_1 = 2.375, \quad x_2 = 12.39.$$

继续迭代下去已经没有必要，因为结果显然会越来越大，不可能趋于某个极限. 这种不收敛的迭代过程称作是**发散**的. 一个发散的迭代过程，纵使进行了千百次迭代，其结果也是毫无价值的.

例 3 表明原方程化为(2.1)的形式不同，有的收敛，有的发散，只有收敛的迭代过程(2.2)才有意义，为此我们首先要研究 $\varphi(x)$ 的不动点的存在性及迭代法(2.2)的收敛性.

7.2.2 不动点的存在性与迭代法的收敛性

首先考察 $\varphi(x)$ 在 $[a,b]$ 上不动点的存在唯一性.

定理 1 设 $\varphi(x)\in C[a,b]$ 满足以下两个条件：

1° 对任意 $x\in[a,b]$ 有 $a\leqslant\varphi(x)\leqslant b$.

2° 存在正常 $L<1$，使对任意 $x,y\in[a,b]$ 都有

$$|\varphi(x)-\varphi(y)|\leqslant L|x-y|. \tag{2.4}$$

则 $\varphi(x)$ 在 $[a,b]$ 上存在唯一的不动点 x^*.

证明 先证不动点存在性. 若 $\varphi(a)=a$ 或 $\varphi(b)=b$，显然 $\varphi(x)$ 在 $[a,b]$ 上存在不动点. 因 $a\leqslant\varphi(x)\leqslant b$，以下设 $\varphi(a)>a$ 及 $\varphi(b)<b$，定义函数

$$f(x)=\varphi(x)-x.$$

显然 $f(x)\in C[a,b]$，且满足 $f(a)=\varphi(a)-a>0$，$f(b)=\varphi(b)-b<0$，由连续函数性质可知存在 $x^*\in(a,b)$ 使 $f(x^*)=0$，即 $x^*=\varphi(x^*)$，x^* 即为 $\varphi(x)$ 的不动点.

再证唯一性. 设 x_1^* 及 $x_2^*\in[a,b]$ 都是 $\varphi(x)$ 的不动点，则由(2.4)得

$$\begin{aligned}|x_1^*-x_2^*|&=|\varphi(x_1^*)-\varphi(x_2^*)|\\&\leqslant L|x_1^*-x_2^*|<|x_1^*-x_2^*|.\end{aligned}$$

引出矛盾. 故 $\varphi(x)$ 的不动点只能是唯一的. 证毕.

在 $\varphi(x)$ 的不动点存在唯一的情况下，可得到迭代法(2.2)收敛的一个充分条件.

定理 2　设 $\varphi(x)\in C[a,b]$ 满足定理 1 中的两个条件，则对任意 $x_0\in[a,b]$，由(2.2)得到的迭代序列 $\{x_k\}$ 收敛到 $\varphi(x)$ 的不动点 x^*，并有误差估计

$$|x_k-x^*|\leqslant\frac{L^k}{1-L}|x_1-x_0|. \tag{2.5}$$

证明　设 $x^*\in[a,b]$ 是 $\varphi(x)$ 在 $[a,b]$ 上的唯一不动点，由条件 1°，可知 $\{x_k\}\in[a,b]$，再由(2.4)得

$$\begin{aligned}|x_k-x^*|&=|\varphi(x_{k-1})-\varphi(x^*)|\\&\leqslant L|x_{k-1}-x^*|\leqslant\cdots\leqslant L^k|x_0-x^*|.\end{aligned}$$

因 $0<L<1$，故当 $k\to\infty$ 时序列 $\{x_k\}$ 收敛到 x^*.

下面再证明估计式(2.5)，由(2.4)有

$$|x_{k+1}-x_k|=|\varphi(x_k)-\varphi(x_{k-1})|\leqslant L|x_k-x_{k-1}|. \tag{2.6}$$

据此反复递推得

$$|x_{k+1}-x_k|\leqslant L^k|x_1-x_0|.$$

于是对任意正整数 p 有

$$\begin{aligned}|x_{k+p}-x_k|&\leqslant|x_{k+p}-x_{k+p-1}|+|x_{k+p-1}-x_{k+p-2}|\\&\quad+\cdots+|x_{k+1}-x_k|\\&\leqslant(L^{k+p-1}+L^{k+p-2}+\cdots+L^k)|x_1-x_0|\\&\leqslant\frac{L^k}{1-L}|x_1-x_0|.\end{aligned}$$

在上式令 $p\to\infty$，注意到 $\lim\limits_{p\to\infty}x_{k+p}=x^*$ 即得式(2.5). 证毕.

迭代过程是个极限过程. 在用迭代法进行实际计算时，必须按精度要求控制迭代次数. 误差估计式(2.5)原则上可用于确定迭代次数，但它由于含有信息 L 而不便于实际应用. 根据式(2.6)，对任意正整数 p 有

$$|x_{k+p}-x_k|\leqslant(L^{p-1}+L^{p-2}+\cdots+1)|x_{k+1}-x_k|$$

$$\leqslant \frac{1}{1-L}\mid x_{k+1}-x_k\mid.$$

在上式中令 $p\to\infty$ 知

$$\mid x^*-x_k\mid\leqslant \frac{1}{1-L}\mid x_{k+1}-x_k\mid.$$

由此可见,只要相邻两次计算结果的偏差 $|x_{k+1}-x_k|$ 足够小即可保证近似值 x_k 具有足够精度.

对定理 1 和定理 2 中的条件 2°,在使用时如果 $\varphi(x)\in C^1[a,b]$ 且对任意 $x\in[a,b]$ 有

$$\mid\varphi'(x)\mid\leqslant L<1, \tag{2.7}$$

则由中值定理可知对 $\forall\, x,y\in[a,b]$ 有

$$\mid\varphi(x)-\varphi(y)\mid=\mid\varphi'(\xi)(x-y)\mid\leqslant L\mid x-y\mid,\ \xi\in(a,b).$$

它表明定理中的条件 2°可用(2.7)代替.

在例 3 中,当 $\varphi(x)=\sqrt[3]{x+1}$ 时,$\varphi'(x)=\frac{1}{3}(x+1)^{-2/3}$,在区间 $[1,2]$ 中,$|\varphi'(x)|\leqslant\frac{1}{3}\left(\frac{1}{4}\right)^{1/3}<1$,故(2.7)成立. 又因 $1\leqslant\sqrt[3]{2}\leqslant\varphi(x)\leqslant\sqrt[3]{3}\leqslant2$,故定理 1 中条件 1°也成立. 所以迭代法是收敛的. 而当 $\varphi(x)=x^3-1$ 时,$\varphi'(x)=3x^2$ 在区间 $[1,2]$ 中 $|\varphi'(x)|>1$ 不满足定理条件.

7.2.3　局部收敛性与收敛阶

上面给出了迭代序列 $\{x_k\}$ 在区间 $[a,b]$ 上的收敛性,通常称为全局收敛性. 有时不易检验定理的条件,实际应用时通常只在不动点 x^* 的邻近考察其收敛性,即局部收敛性.

定义 1　设 $\varphi(x)$ 有不动点 x^*,如果存在 x^* 的某个邻域 $R:|x-x^*|\leqslant\delta$,对任意 $x_0\in R$,迭代(2.2)产生的序列 $\{x_k\}\in R$,且收敛到 x^*,则称迭代法(2.2)**局部收敛**.

定理 3　设 x^* 为 $\varphi(x)$ 的不动点,$\varphi'(x)$ 在 x^* 的某个邻域连

续,且$|\varphi'(x^*)|<1$,则迭代法(2.2)局部收敛.

证明　由连续函数的性质,存在 x^* 的某个邻域 $R: |x-x^*|\leqslant\delta$,使对于任意 $x\in R$ 成立

$$|\varphi'(x)|\leqslant L<1.$$

此外,对于任意 $x\in R$,总有 $\varphi(x)\in R$,这是因为

$$|\varphi(x)-x^*|=|\varphi(x)-\varphi(x^*)|\leqslant L|x-x^*|\leqslant|x-x^*|.$$

于是依据定理 2 可以断定迭代过程 $x_{k+1}=\varphi(x_k)$ 对于任意初值 $x_0\in R$ 均收敛. 证毕.

下面讨论迭代序列的收敛速度问题,先看例.

例 4　用不同方法求方程 $x^2-3=0$ 的根 $x^*=\sqrt{3}$.

解　这里 $f(x)=x^2-3$,可改写为各种不同的等价形式 $x=\varphi(x)$,其不动点为 $x^*=\sqrt{3}$. 由此构造不同的迭代法:

(1) $x_{k+1}=x_k^2+x_k-3$, $\varphi(x)=x^2+x-3$, $\varphi'(x)=2x+1$,

$\varphi'(x^*)=\varphi'(\sqrt{3})=2\sqrt{3}+1>1$.

(2) $x_{k+1}=\dfrac{3}{x_k}$, $\varphi(x)=\dfrac{3}{x}$, $\varphi'(x)=-\dfrac{3}{x^2}$, $\varphi'(x^*)=-1$.

(3) $x_{k+1}=x_k-\dfrac{1}{4}(x_k^2-3)$, $\varphi(x)=x-\dfrac{1}{4}(x^2-3)$,

$\varphi'(x)=1-\dfrac{1}{2}x$,

$\varphi'(x^*)=1-\dfrac{\sqrt{3}}{2}\approx 0.134<1$.

(4) $x_{k+1}=\dfrac{1}{2}\left(x_k+\dfrac{3}{x_k}\right)$, $\varphi(x)=\dfrac{1}{2}\left(x+\dfrac{3}{x}\right)$,

$\varphi'(x)=\dfrac{1}{2}\left(1-\dfrac{3}{x^2}\right)$, $\varphi'(x^*)=\varphi'(\sqrt{3})=0$.

取 $x_0=2$,对上述 4 种迭代法,计算三步所得的结果如下表.

表 7-3 计算结果

k	x_k	迭代法(1)	迭代法(2)	迭代法(3)	迭代法(4)
0	x_0	2	2	2	2
1	x_1	3	1.5	1.75	1.75
2	x_2	9	2	1.73475	1.732143
3	x_3	87	1.5	1.732361	1.732051
⋮	⋮	⋮	⋮	⋮	⋮

注意$\sqrt{3}=1.7320508\cdots$，从计算结果看到迭代法(1)及(2)均不收敛，且它们均不满足定理 3 中的局部收敛条件，迭代法(3)和(4)均满足局部收敛条件，且迭代法(4)比(3)收敛快，因在迭代法(4)中 $\varphi'(x^*)=0$. 为了衡量迭代法(2.2)收敛速度的快慢可给出以下定义.

定义 2 设迭代过程 $x_{k+1}=\varphi(x_k)$ 收敛于方程 $x=\varphi(x)$ 的根 x^*，如果迭代误差 $e_k=x_k-x^*$ 当 $k\to\infty$ 时成立下列渐近关系式

$$\frac{e_{k+1}}{e_k^p}\to C \qquad (\text{常数 } C\neq 0),$$

则称该迭代过程是 **p 阶收敛**的. 特别地，$p=1$ 时称**线性收敛**，$p>1$ 时称**超线性收敛**，$p=2$ 时称**平方收敛**.

定理 4 对于迭代过程 $x_{k+1}=\varphi(x_k)$，如果 $\varphi^{(p)}(x)$ 在所求根 x^* 的邻近连续，并且

$$\varphi'(x^*)=\varphi''(x^*)=\cdots=\varphi^{(p-1)}(x^*)=0,$$
$$\varphi^{(p)}(x^*)\neq 0, \tag{2.8}$$

则该迭代过程在点 x^* 邻近是 p 阶收敛的.

证明 由于 $\varphi'(x^*)=0$，据定理 3 立即可以断定迭代过程 $x_{k+1}=\varphi(x_k)$ 具有局部收敛性.

再将 $\varphi(x_k)$ 在根 x^* 处做泰勒展开，利用条件(2.8)，则有

$$\varphi(x_k) = \varphi(x^*) + \frac{\varphi^{(p)}(\xi)}{p!}(x_k - x^*)^p, \xi \text{ 在 } x_k \text{ 与 } x^* \text{ 之间.}$$

注意到 $\varphi(x_k)=x_{k+1}, \varphi(x^*)=x^*$，由上式得

$$x_{k+1} - x^* = \frac{\varphi^{(p)}(\xi)}{p!}(x_k - x^*)^p,$$

因此对迭代误差，当 $k\to\infty$ 时有

$$\frac{e_{k+1}}{e_k^p} \to \frac{\varphi^{(p)}(x^*)}{p!}. \tag{2.9}$$

这表明迭代过程 $x_{k+1}=\varphi(x_k)$ 确实为 p 阶收敛. 证毕.

上述定理告诉我们，迭代过程的收敛速度依赖于迭代函数 $\varphi(x)$ 的选取. 如果当 $x\in[a,b]$ 时 $\varphi'(x)\neq 0$，则该迭代过程只可能是线性收敛.

在例 4 中，迭代法(3)的 $\varphi'(x^*)\neq 0$，故它只是线性收敛，而迭代法(4)的 $\varphi'(x^*)=0$，而 $\varphi''(x)=\dfrac{6}{x^3}, \varphi''(x^*)=\dfrac{2}{\sqrt{3}}\neq 0$. 由定理 4 知 $p=2$，即该迭代过程为 2 阶收敛.

7.3 迭代收敛的加速方法

7.3.1 埃特金加速收敛方法

对于收敛的迭代过程，只要迭代足够多次，就可以使结果达到任意的精度，但有时迭代过程收敛缓慢，从而使计算量变得很大，因此迭代过程的加速是个重要的课题.

设 x_0 是根 x^* 的某个近似值，用迭代公式校正一次得

$$x_1 = \varphi(x_0),$$

而由微分中值定理，有

$$x_1 - x^* = \varphi(x_0) - \varphi(x^*) = \varphi'(\xi)(x_0 - x^*),$$

其中 ξ 介于 x^* 与 x_0 之间.

假定 $\varphi'(x)$改变不大，近似地取某个近似值 L，则有

$$x_1 - x^* \approx L(x_0 - x^*). \tag{3.1}$$

若将校正值 $x_1=\varphi(x_0)$再校正一次，又得

$$x_2 = \varphi(x_1),$$

由于

$$x_2 - x^* \approx L(x_1 - x^*),$$

将它与(3.1)式联立，消去未知的 L，有

$$\frac{x_1 - x^*}{x_2 - x^*} \approx \frac{x_0 - x^*}{x_1 - x^*}.$$

由此推知

$$x^* \approx \frac{x_0 x_2 - x_1^2}{x_2 - 2x_1 + x_0} = x_0 - \frac{(x_1 - x_0)^2}{x_2 - 2x_1 + x_0}.$$

在计算了 x_1 及 x_2 之后，可用上式右端作为 x^* 的新近似，记作 $\bar{x}_1$. 一般情形是由 x_k 计算 x_{k+1}, x_{k+2}，记

$$\begin{aligned}\bar{x}_{k+1} &= x_k - \frac{(x_{k+1} - x_k)^2}{x_k - 2x_{k+1} + x_{k+2}} \\ &= x_k - (\Delta x_k)^2/\Delta^2 x_k \qquad (k = 0,1,\cdots).\end{aligned} \tag{3.2}$$

(3.2)称为埃特金(Aitken)Δ^2 加速方法.

可以证明

$$\lim_{k\to\infty} \frac{\bar{x}_{k+1} - x^*}{x_k - x^*} = 0.$$

它表明序列$\{\bar{x}_k\}$的收敛速度比$\{x_k\}$的收敛速度快.

7.3.2 斯蒂芬森迭代法

埃特金方法不管原序列$\{x_k\}$是怎样产生的，对$\{x_k\}$进行加速计算，得到序列$\{\bar{x}_k\}$. 如果把埃特金加速技巧与不动点迭代结合，则可得到如下的迭代法：

$$y_k = \varphi(x_k), \quad z_k = \varphi(y_k),$$

$$x_{k+1} = x_k - \frac{(y_k - x_k)^2}{z_k - 2y_k + x_k} \qquad (k = 0,1,\cdots), \tag{3.3}$$

称为斯蒂芬森(Steffensen)迭代法. 它可以这样理解,我们要求 $x=\varphi(x)$ 的根 x^*,令 $\varepsilon(x)=\varphi(x)-x$,$\varepsilon(x^*)=\varphi(x^*)-x^*=0$,已知 x^* 的近似值 x_k 及 y_k,其误差分别为

$$\varepsilon(x_k)=\varphi(x_k)-x_k=y_k-x_k,$$

$$\varepsilon(y_k)=\varphi(y_k)-y_k=z_k-y_k.$$

把误差 $\varepsilon(x)$"外推到零",即过 $(x_k,\varepsilon(x_k))$ 及 $(y_k,\varepsilon(y_k))$ 两点做线性插值函数,它与 x 轴交点就是(3.3)中的 x_{k+1},即方程

$$\varepsilon(x_k)+\frac{\varepsilon(y_k)-\varepsilon(x_k)}{y_k-x_k}(x-x_k)=0$$

的解

$$x=x_k-\frac{\varepsilon(x_k)}{\varepsilon(y_k)-\varepsilon(x_k)}(y_k-x_k)=x_k-\frac{(y_k-x_k)^2}{z_k-2y_k+x_k}=x_{k+1}.$$

实际上(3.3)是将不动点迭代法(2.2)计算两步合并成一步得到的,可将它写成另一种不动点迭代

$$x_{k+1}=\psi(x_k)\qquad(k=0,1,\cdots),\tag{3.4}$$

其中

$$\psi(x)=x-\frac{[\varphi(x)-x]^2}{\varphi(\varphi(x))-2\varphi(x)+x}.\tag{3.5}$$

对不动点迭代(3.4)有以下局部收敛性定理.

定理 5　若 x^* 为(3.5)定义的迭代函数 $\psi(x)$ 的不动点,则 x^* 为 $\varphi(x)$ 的不动点. 反之,若 x^* 为 $\varphi(x)$ 的不动点,设 $\varphi''(x)$ 存在,$\varphi'(x^*)\neq1$,则 x^* 是 $\psi(x)$ 的不动点,且斯蒂芬森迭代法(3.3)是 2 阶收敛的.

证明可见[2].

例 5　用斯蒂芬森迭代法求解方程(2.3).

解　例 3 中已指出下列迭代

$$x_{k+1}=x_k^3-1$$

是发散的,现用(3.3)计算,取 $\varphi(x)=x^3-1$,计算结果如下表.

表 7-4 计算结果

k	x_k	y_k	z_k
0	1.5	2.37500	12.3965
1	1.41629	1.84092	5.23888
2	1.35565	1.49140	2.31728
3	1.32895	1.34710	1.44435
4	1.32480	1.32518	1.32714
5	1.32472		

计算表明它是收敛的，这说明即使迭代法(2.2)不收敛，用斯蒂芬森迭代法(3.3)仍可能收敛. 至于原来已收敛的迭代法(2.2)，由定理 5 可知它可达到 2 阶收敛. 更进一步还可知若(2.2)为 p 阶收敛，则(3.3)为 $p+1$ 阶收敛(见[2]).

例 6 求方程 $3x^2-e^x=0$ 在$[3,4]$中的解.

解 由方程得 $e^x=3x^2$，取对数得

$$x = \ln 3x^2 = 2\ln x + \ln 3 = \varphi(x).$$

若构造迭代法

$$x_{k+1} = 2\ln x_k + \ln 3,$$

由于 $\varphi'(x)=\dfrac{2}{x}$，$\max\limits_{3\leqslant x\leqslant 4}|\varphi'(x)|\leqslant\dfrac{2}{3}<1$，且当 $x\in[3,4]$时，$\varphi(x)\in[3,4]$，根据定理 2 此迭代法是收敛的. 若取 $x_0=3.5$ 迭代 16 次得 $x_{16}=3.73307$，有六位有效数字.

若用(3.3)进行加速，计算结果如下：

k	x_k	y_k	z_k
0	3.5	3.60414	3.66202
1	3.73444	3.73381	3.73347
2	3.73307		

表 7-4 计算结果

k	x_k	y_k	z_k
0	1.5	2.37500	12.3965
1	1.41629	1.84092	5.23888
2	1.35565	1.49140	2.31728
3	1.32895	1.34710	1.44435
4	1.32480	1.32518	1.32714
5	1.32472		

计算表明它是收敛的，这说明即使迭代法(2.2)不收敛，用斯蒂芬森迭代法(3.3)仍可能收敛．至于原来已收敛的迭代法(2.2)，由定理 5 可知它可达到 2 阶收敛．更进一步还可知若(2.2)为 p 阶收敛，则(3.3)为 $p+1$ 阶收敛(见[2])．

例 6 求方程 $3x^2-e^x=0$ 在[3,4]中的解．

解 由方程得 $e^x=3x^2$，取对数得

$$x=\ln 3x^2=2\ln x+\ln 3=\varphi(x).$$

若构造迭代法

$$x_{k+1}=2\ln x_k+\ln 3,$$

由于 $\varphi'(x)=\frac{2}{x}$，$\max\limits_{3\leqslant x\leqslant 4}|\varphi'(x)|\leqslant\frac{2}{3}<1$，且当 $x\in[3,4]$时，$\varphi(x)\in[3,4]$，根据定理 2 此迭代法是收敛的．若取 $x_0=3.5$ 迭代 16 次得 $x_{16}=3.73307$，有六位有效数字．

若用(3.3)进行加速，计算结果如下：

k	x_k	y_k	z_k
0	3.5	3.60414	3.66202
1	3.73444	3.73381	3.73347
2	3.73307		

$$y = f(x_k) + f'(x_k)(x - x_k).$$

这样求得的值 x_{k+1} 必满足(4.1)，从而就是牛顿公式(4.2)的计算结果.由于这种几何背景，牛顿法亦称**切线法**.

关于牛顿法(4.2)的收敛性，可直接由定理 4 得到，对(4.2)其迭代函数为

$$\varphi(x) = x - \frac{f(x)}{f'(x)},$$

由于
$$\varphi'(x) = \frac{f(x)f''(x)}{[f'(x)]^2}.$$

假定 x^* 是 $f(x)$ 的一个单根，即 $f(x^*)=0, f'(x^*)\neq 0$，则由上式知 $\varphi'(x^*)=0$，于是依据定理 4 可以断定，牛顿法在根 x^* 的邻近是平方收敛的.又因

$\varphi''(x^*)=\dfrac{f''(x^*)}{f'(x^*)}$，故由(2.9)可得

$$\lim_{k\to\infty}\frac{x_{k+1}-x^*}{(x_k-x^*)^2} = \frac{f''(x^*)}{2f'(x^*)}. \tag{4.3}$$

例 7 用牛顿法解方程

$$xe^x - 1 = 0. \tag{4.4}$$

解 这里牛顿公式为

$$x_{k+1} = x_k - \frac{x_k - e^{-x_k}}{1 + x_k},$$

取迭代初值 $x_0=0.5$，迭代结果列于表 7-5 中.

表 7-5 计算结果

k	x_k
0	0.5
1	0.57102
2	0.56716
3	0.56714

所给方程(4.4)实际上是方程 $x=e^{-x}$ 的等价形式.若用不动点迭代到同一精度要迭代 17 次，可见牛顿法的收敛速度是很快的.

下面列出牛顿法的计算

步骤：

步骤1 准备 选定初始近似值 x_0，计算 $f_0=f(x_0)$，$f_0'=f'(x_0)$.

步骤2 迭代 按公式

$$x_1=x_0-f_0/f_0'$$

迭代一次，得新的近似值 x_1，计算 $f_1=f(x_1)$，$f_1'=f'(x_1)$.

步骤3 控制 如果 x_1 满足 $|\delta|<\varepsilon_1$ 或 $|f_1|<\varepsilon_2$，则终止迭代，以 x_1 作为所求的根；否则转步骤4. 此处 $\varepsilon_1,\varepsilon_2$ 是允许误差，而

$$\delta=\begin{cases}|x_1-x_0|, & \text{当 } |x_1|<C \text{ 时;}\\ \dfrac{|x_1-x_0|}{|x_1|}, & \text{当 } |x_1|\geqslant C \text{ 时,}\end{cases}$$

其中 C 是取绝对误差或相对误差的控制常数，一般可取 $C=1$.

步骤4 修改 如果迭代次数达到预先指定的次数 N，或者 $f_1'=0$，则方法失败；否则以 (x_1,f_1,f_1') 代替 (x_0,f_0,f_0') 转步骤2继续迭代.

7.4.2 牛顿法应用举例

对于给定的正数 C，应用牛顿法解二次方程

$$x^2-C=0,$$

可导出求开方值 $\sqrt{C}$ 的计算程序

$$x_{k+1}=\frac{1}{2}\left(x_k+\frac{C}{x_k}\right). \tag{4.5}$$

我们现在证明，这种迭代公式对于任意初值 $x_0>0$ 都是收敛的.

事实上，对(4.5)式施行配方手续，易知

$$x_{k+1}-\sqrt{C}=\frac{1}{2x_k}(x_k-\sqrt{C})^2;$$

$$x_{k+1}+\sqrt{C}=\frac{1}{2x_k}(x_k+\sqrt{C})^2.$$

以上两式相除得

$$\frac{x_{k+1}-\sqrt{C}}{x_{k+1}+\sqrt{C}}=\left(\frac{x_k-\sqrt{C}}{x_k+\sqrt{C}}\right)^2.$$

据此反复递推有

$$\frac{x_k-\sqrt{C}}{x_k+\sqrt{C}}=\left(\frac{x_0-\sqrt{C}}{x_0+\sqrt{C}}\right)^{2^k}, \tag{4.6}$$

记

$$q=\frac{x_0-\sqrt{C}}{x_0+\sqrt{C}},$$

整理(4.6)式,得

$$x_k-\sqrt{C}=2\sqrt{C}\,\frac{q^{2^k}}{1-q^{2^k}}.$$

对任意 $x_0>0$,总有 $|q|<1$,故由上式推知,当 $k\to\infty$ 时 $x_k\to\sqrt{C}$,即迭代过程恒收敛.

例 8 求 $\sqrt{115}$.

解 取初值 $x_0=10$,对 $C=115$ 按(4.5)式迭代 3 次便得到精度为 10^{-6} 的结果(见表 7-6).

表 7-6 计算结果

k	x_k
0	10
1	10.750000
2	10.723837
3	10.723805
4	10.723805

由于公式(4.5)对任意初值 $x_0>0$ 均收敛,并且收敛的速度很快,因此我们可取确定的初值如 $x_0=1$ 编制通用程序.用这个通用程序求 $\sqrt{115}$,也只要迭代 7 次便得到了上面的结果 10.723805.

7.4.3 简化牛顿法与牛顿下山法

牛顿法的优点是收敛快,缺点一是每步迭代要计算 $f(x_k)$ 及

$f'(x_k)$，计算量较大且有时 $f'(x_k)$ 计算较困难，二是初始近似 x_0 只在根 x^* 附近才能保证收敛，如 x_0 给的不合适可能不收敛. 为克服这两个缺点，通常可用下述方法.

(1) 简化牛顿法，也称平行弦法. 其迭代公式为

$$x_{k+1} = x_k - Cf(x_k) \qquad C \neq 0,\ k = 0,1,\cdots. \tag{4.7}$$

迭代函数 $\varphi(x) = x - Cf(x)$.

若 $|\varphi'(x)| = |1 - Cf'(x)| < 1$，即取 $0 < Cf'(x) < 2$. 在根 x^* 附近成立，则迭代法(4.7)局部收敛.

在(4.7)中取 $C = \dfrac{1}{f'(x_0)}$，则称为**简化牛顿法**，这类方法计算量省，但只有线性收敛，其几何意义是用平行弦与 x 轴交点作为 x^* 的近似. 如图 7-4 所示.

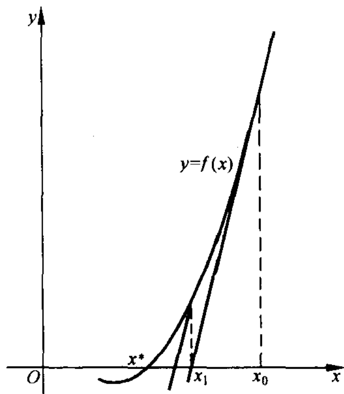


图 7-4

(2) 牛顿下山法. 牛顿法收敛性依赖初值 x_0 的选取. 如果 x_0 偏离所求根 x^* 较远，则牛顿法可能发散.

例如，用牛顿法求解方程

$$x^3 - x - 1 = 0. \tag{4.8}$$

此方程在 $x = 1.5$ 附近的一个根 x^*. 设取迭代初值 $x_0 = 1.5$，用牛顿法公式

$$x_{k+1} = x_k - \frac{x_k^3 - x_k - 1}{3x_k^2 - 1} \tag{4.9}$$

计算得

$$x_1 = 1.34783, \quad x_2 = 1.32520, \quad x_3 = 1.32472.$$

迭代 3 次得到的结果 x_3 有 6 位有效数字.

但是，如果改用 $x_0 = 0.6$ 作为迭代初值，则依牛顿法公式

(4.9)迭代一次得

$$x_1 = 17.9.$$

这个结果反而比 $x_0=0.6$ 更偏离了所求的根 $x^*=1.32472$.

为了防止迭代发散,我们对迭代过程再附加一项要求,即具有单调性:

$$|f(x_{k+1})|<|f(x_k)|. \tag{4.10}$$

满足这项要求的算法称**下山法**.

我们将牛顿法与下山法结合起来使用,即在下山法保证函数值稳定下降的前提下,用牛顿法加快收敛速度.为此,我们将牛顿法的计算结果

$$\bar{x}_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

与前一步的近似值 x_k 适当加权平均作为新的改进值

$$x_{k+1} = \lambda\bar{x}_{k+1} + (1-\lambda)x_k, \tag{4.11}$$

其中 $\lambda(0<\lambda\leqslant 1)$称为下山因子,(4.11)即为

$$x_{k+1} = x_k - \lambda\frac{f(x_k)}{f'(x_k)} \quad (k=0,1,\cdots), \tag{4.12}$$

称为**牛顿下山法**.选择下山因子时从 $\lambda=1$ 开始,逐次将 λ 减半进行试算,直到能使下降条件(4.10)成立为止.若用此法解方程(4.8),当 $x_0=0.6$ 时由(4.9)求得 $x_1=17.9$,它不满足条件(4.10),通过 λ 逐次取半进行试算,当 $\lambda=\frac{1}{32}$ 时可求得 $x_1=1.140625$.此时有 $f(x_1)=-0.656643$,而 $f(x_0)=-1.384$,显然 $|f(x_1)|<|f(x_0)|$.由 x_1 计算 $x_2,x_3,\cdots$时 $\lambda=1$,均能使条件(4.10)成立.计算结果如下:

$x_2=1.36181$, $f(x_2)=0.1866$;

$x_3=1.32628$, $f(x_3)=0.00667$;

$x_4=1.32472$, $f(x_4)=0.0000086$.

x_4 即为 x^* 的近似. 一般情况只要能使条件(4.10)成立,则可得到 $\lim\limits_{k\to\infty} f(x_k)=0$,从而使$\{x_k\}$收敛.

7.4.4 重根情形

设 $f(x)=(x-x^*)^m g(x)$,整数 $m\geqslant 2$,$g(x^*)\neq 0$,则 x^* 为方程 $f(x)=0$ 的 m 重根,此时有

$$f(x^*)=f'(x^*)=\cdots=f^{(m-1)}(x^*)=0,\ f^{(m)}(x^*)\neq 0.$$

只要 $f'(x_k)\neq 0$ 仍可用牛顿法(4.2)计算,此时迭代函数 $\varphi(x)=x-\dfrac{f(x)}{f'(x)}$的导数为 $\varphi'(x^*)=1-\dfrac{1}{m}\neq 0$,且$|\varphi'(x^*)|<1$,所以牛顿法求重根只是线性收敛. 若取

$$\varphi(x)=x-m\frac{f(x)}{f'(x)},$$

则 $\varphi'(x^*)=0$. 用迭代法

$$x_{k+1}=x_k-m\frac{f(x_k)}{f'(x_k)}\qquad(k=0,1,\cdots)\tag{4.13}$$

求 m 重根,则具有 2 阶收敛,但要知道 x^* 的重数 m.

构造求重根的迭代法,还可令 $\mu(x)=f(x)/f'(x)$,若 x^* 是 $f(x)=0$的 m 重根,则

$$\mu(x)=\frac{(x-x^*)g(x)}{mg(x)+(x-x^*)g'(x)},$$

故 x^* 是 $\mu(x)=0$ 的单根. 对它用牛顿法,其迭代函数为

$$\varphi(x)=x-\frac{\mu(x)}{\mu'(x)}=x-\frac{f(x)f'(x)}{[f'(x)]^2-f(x)f''(x)}.$$

从而可构造迭代法

$$x_{k+1}=x_k-\frac{f(x_k)f'(x_k)}{[f'(x_k)]^2-f(x_k)f''(x_k)}\quad(k=0,1,\cdots),\tag{4.14}$$

它是二阶收敛的.

例 9 方程 $x^4-4x^2+4=0$ 的根 $x^*=\sqrt{2}$ 是二重根，用上述三种方法求根.

解 先求出三种方法的迭代公式：

(1) 牛顿法 $x_{k+1}=x_k-\dfrac{x_k^2-2}{4x_k}$.

(2) 用(4.13)式 $x_{k+1}=x_k-\dfrac{x_k^2-2}{2x_k}$.

(3) 用(4.14)式 $x_{k+1}=x_k-\dfrac{x_k(x_k^2-2)}{x_k^2+2}$.

取初值 $x_0=1.5$，计算结果如表 7-7.

表 7-7 三种方法数值结果

k	x_k	方法(1)	方法(2)	方法(3)
1	x_1	1.458333333	1.416666667	1.411764706
2	x_2	1.436607143	1.414215686	1.414211438
3	x_3	1.425497619	1.414213562	1.414213562

计算三步，方法(2)及(3)均达到 10 位有效数字，而用牛顿法只有线性收敛，要达到同样精度需迭代 30 次.

7.5 弦截法与抛物线法

用牛顿法求方程(1.1)的根，每步除计算 $f(x_k)$ 外还要算 $f'(x_k)$，当函数 $f(x)$ 比较复杂时，计算 $f'(x)$ 往往较困难，为此可以利用已求函数值 $f(x_k),f(x_{k-1}),\cdots$ 来回避导数值 $f'(x_k)$ 的计算. 这类方法是建立在插值原理基础上的，下面介绍两种常用方法.

7.5.1 弦截法

设 x_k,x_{k-1} 是 $f(x)=0$ 的近似根，我们利用 $f(x_k),f(x_{k-1})$ 构

造一次插值多项式 $p_1(x)$，并用 $p_1(x)=0$ 的根作为 $f(x)=0$ 的新的近似根 x_{k+1}. 由于

$$p_1(x) = f(x_k) + \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}(x - x_k). \tag{5.1}$$

因此有

$$x_{k+1} = x_k - \frac{f(x_k)}{f(x_k) - f(x_{k-1})}(x_k - x_{k-1}). \tag{5.2}$$

这样导出的迭代公式(5.2)可以看做牛顿公式

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

中的导数 $f'(x_k)$ 用差商 $\frac{f(x_k)-f(x_{k-1})}{x_k-x_{k-1}}$ 取代的结果.

现在解释这种迭代过程的几何意义. 如图 7-5，曲线 $y=f(x)$ 上横坐标为 x_k, x_{k-1} 的点分别记为 P_k, P_{k-1}，则弦线 $\overline{P_kP_{k-1}}$ 的斜率等于差商值 $\frac{f(x_k)-f(x_{k-1})}{x_k-x_{k-1}}$，其方程是

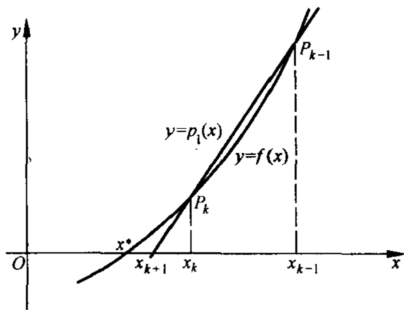


图 7-5

$$y=f(x_k)+\frac{f(x_k)-f(x_{k-1})}{x_k-x_{k-1}}(x-x_k).$$

因之，按(5.2)式求得的 x_{k+1} 实际上是弦线 $\overline{P_kP_{k-1}}$ 与 x 轴交点的横坐标. 这种算法因此而称为**弦截法**.

弦截法与切线法(牛顿法)都是线性化方法,但两者有本质的区别.切线法在计算 x_{k+1} 时只用到前一步的值 x_k,而弦截法(5.2),在求 x_{k+1} 时要用到前面两步的结果 x_k,x_{k-1},因此使用这种方法必须先给出两个开始值 x_0,x_1.

例 10 用弦截法解方程

$$f(x)=xe^x-1=0.$$

解 设取 $x_0=0.5,x_1=0.6$ 作为开始值,用弦截法求得的结果见表 7-8,比较例 7 牛顿法的计算结果可以看出,弦截法的收敛速度也是相当快的.

表 7-8 计算结果

k	x_k
0	0.5
1	0.6
2	0.56532
3	0.56709
4	0.56714

实际上,下述定理断言,弦截法具有超线性的收敛性.

定理 6 假设 $f(x)$ 在根 x^* 的邻域 $\Delta:|x-x^*|\leqslant\delta$ 内具有二阶连续导数,且对任意 $x\in\Delta$ 有 $f'(x)\neq0$,又初值 $x_0,x_1\in\Delta$,那么当邻域 Δ 充分小时,弦截法(5.2)将按阶 $p=\frac{1+\sqrt{5}}{2}\approx1.618$ 收敛到根 x^*.这里 p 是方程 $\lambda^2-\lambda-1=0$ 的正根.

定理证明可见[2].

7.5.2 抛物线法

设已知方程 $f(x)=0$ 的三个近似根 x_k,x_{k-1},x_{k-2},我们以这三点为节点构造二次插值多项式 $p_2(x)$,并适当选取 $p_2(x)$ 的一个零点 x_{k+1} 作为新的近似根,这样确定的迭代过程称**抛物线法**,亦称**密勒**(Müller)**法**.在几何图形上,这种方法的基本思想是用抛物线 $y=p_2(x)$ 与 x 轴的交点 x_{k+1} 作为所求根 x^* 的近似位置(图 7-6).

现在推导抛物线法的计算公式.插值多项式

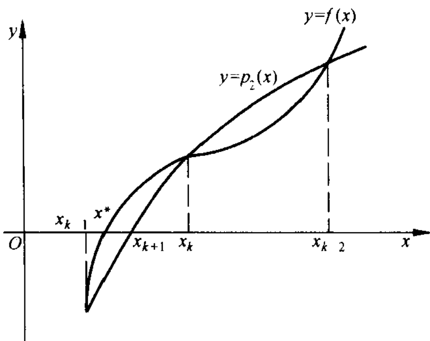


图 7-6

$$p_2(x)=f(x_k)+f[x_k,x_{k-1}](x-x_k)$$
$$+f[x_k,x_{k-1},x_{k-2}](x-x_k)(x-x_{k-1}).$$

有两个零点：

$$x_{k+1}=x_k-\frac{2f(x_k)}{\omega\pm\sqrt{\omega^2-4f(x_k)f[x_k,x_{k-1},x_{k-2}]}},\qquad(5.3)$$

式中

$$\omega=f[x_k,x_{k-1}]+f[x_k,x_{k-1},x_{k-2}](x_k-x_{k-1}).$$

为了从(5.3)式定出一个值 x_{k+1}，我们需要讨论根式前正负号的取舍问题.

在 x_k,x_{k-1},x_{k-2} 三个近似根中，自然假定 x_k 更接近所求的根 x^*，这时，为了保证精度，我们选式 (5.3)中较接近 x_k 的一个值作为新的近似根 x_{k+1}. 为此，只要取根式前的符号与 ω 的符号相同.

例 11 用抛物线法求解方程 $f(x)=xe^x-1=0$.

解 设用表 7-8 的前三个值

$$x_0=0.5,\quad x_1=0.6,\quad x_2=0.56532$$

作为开始值，计算得

$f(x_0)=-0.175639$，$f(x_1)=-0.093271$，

$f(x_2)=-0.005031$，

$f[x_1,x_0]=2.68910$，$f[x_2,x_1]=2.83373$，

$f[x_2,x_1,x_0]=2.21418$.

故

$$\omega = f[x_2,x_1]+f[x_2,x_1,x_0](x_2-x_1) = 2.75694.$$

代入(5.3)式求得

$$x_3 = x_2 - \frac{2f(x_2)}{\omega+\sqrt{\omega^2-4f(x_2)f(x_2,x_1,x_0)}} = 0.56714.$$

以上计算表明，抛物线法比弦截法收敛得更快.

事实上，在一定条件下可以证明，对于抛物线法，迭代误差有下列渐近关系式

$$\frac{|e_{k+1}|}{|e_k|^{1.840}} \to \left|\frac{f'''(x^*)}{6f'(x^*)}\right|^{0.42}.$$

可见抛物线法也是超线性收敛的，其收敛的阶 $p=1.840$(是方程 $\lambda^3-\lambda^2-\lambda-1=0$ 的根)，收敛速度比弦截法更接近于牛顿法.

从(5.3)看到，即使 x_{k-2},x_{k-1},x_k 均为实数，x_{k+1} 也可以是复数，所以抛物线法适用于求多项式的实根和复根.

7.6 解非线性方程组的牛顿迭代法

考虑方程组

$$\begin{cases} f_1(x_1,\cdots,x_n) = 0, \\ \cdots\cdots\cdots\cdots\cdots \\ f_n(x_1,\cdots,x_n) = 0. \end{cases} \tag{6.1}$$

其中 $f_1,\cdots,f_n$ 均为$(x_1,\cdots,x_n)$的多元函数. 若用向量记号记 $\boldsymbol{x}=(x_1,\cdots,x_n)^{\mathrm{T}}\in\mathbf{R}^n$，$\boldsymbol{F}=(f_1,\cdots,f_n)^{\mathrm{T}}$，(6.1)就可写成

$$\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{0}. \tag{6.2}$$

当 $n\geqslant 2$，且 $f_i(i=1,\cdots,n)$ 中至少有一个是自变量 $x_i(i=1,\cdots,n)$ 的非线性函数时，则称方程组(6.1)为**非线性方程组**. 非线性方程

组求根问题是前面介绍的方程(即 $n=1$)求根的直接推广,实际上只要把前面介绍的单变量函数 $f(x)$ 看成向量函数 $\boldsymbol{F}(\boldsymbol{x})$ 则可将单变量方程求根方法推广到方程组(6.2). 若已给出方程(6.2)的一个近似根 $\boldsymbol{x}^{(k)}=(x_1^{(k)},\cdots,x_n^{(k)})^{\mathrm{T}}$,将函数 $\boldsymbol{F}(\boldsymbol{x})$ 的分量 $f_i(\boldsymbol{x})$ $(i=1,\cdots,n)$ 在 $\boldsymbol{x}^{(k)}$ 用多元函数泰勒展开,并取其线性部分,则可表示为

$$\boldsymbol{F}(\boldsymbol{x}) \approx \boldsymbol{F}(\boldsymbol{x}^{(k)}) + \boldsymbol{F}'(\boldsymbol{x}^{(k)})(\boldsymbol{x}-\boldsymbol{x}^{(k)}).$$

令上式右端为零,得到线性方程组

$$\boldsymbol{F}'(\boldsymbol{x}^{(k)})(\boldsymbol{x}-\boldsymbol{x}^{(k)}) = -\boldsymbol{F}(\boldsymbol{x}^{(k)}), \tag{6.3}$$

其中

$$\boldsymbol{F}'(\boldsymbol{x}) = \begin{bmatrix} \dfrac{\partial f_1(\boldsymbol{x})}{\partial x_1} & \dfrac{\partial f_1(\boldsymbol{x})}{\partial x_2} & \cdots & \dfrac{\partial f_1(\boldsymbol{x})}{\partial x_n} \\ \dfrac{\partial f_2(\boldsymbol{x})}{\partial x_1} & \dfrac{\partial f_2(\boldsymbol{x})}{\partial x_2} & \cdots & \dfrac{\partial f_2(\boldsymbol{x})}{\partial x_n} \\ \vdots & \vdots & & \vdots \\ \dfrac{\partial f_n(\boldsymbol{x})}{\partial x_1} & \dfrac{\partial f_n(\boldsymbol{x})}{\partial x_2} & \cdots & \dfrac{\partial f_n(\boldsymbol{x})}{\partial x_n} \end{bmatrix} \tag{6.4}$$

称为 $\boldsymbol{F}(\boldsymbol{x})$ 的**雅可比**(Jacobi)**矩阵**. 求解线性方程组(6.3),并记解为 $\boldsymbol{x}^{(k+1)}$,则得

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \boldsymbol{F}'(\boldsymbol{x}^{(k)})^{-1}\boldsymbol{F}(\boldsymbol{x}^{(k)}) \qquad (k=0,1,\cdots). \tag{6.5}$$

这就是解非线性方程组(6.2)的牛顿迭代法.

例 12 求解方程组

$$\begin{cases} f_1(x_1,x_2) = x_1 + 2x_2 - 3 = 0, \\ f_2(x_1,x_2) = 2x_1^2 + x_2^2 - 5 = 0. \end{cases}$$

给定初值 $\boldsymbol{x}^{(0)}=(1.5,1.0)^{\mathrm{T}}$,用牛顿法求解.

解 先求雅可比矩阵

$$\boldsymbol{F}'(\boldsymbol{x}) = \begin{pmatrix} 1 & 2 \\ 4x_1 & 2x_2 \end{pmatrix}, \quad \boldsymbol{F}'(\boldsymbol{x})^{-1} = -\frac{1}{2x_2-8x_1}\begin{pmatrix} 2x_2 & -2 \\ -4x_1 & 1 \end{pmatrix}.$$

由牛顿法(6.5)得

$$x^{(k+1)} = x^{(k)} - \frac{1}{2x_2^{(k)} - 8x_1^{(k)}} \begin{bmatrix} 2x_2^{(k)} & -2 \\ -4x_1^{(k)} & 1 \end{bmatrix} \begin{bmatrix} x_1^{(k)} + 2x_2^{(k)} - 3 \\ 2(x_1^{(k)})^2 + (x_2^{(k)})^2 - 5 \end{bmatrix},$$

即

$$\begin{cases} x_1^{(k+1)} = x_1^{(k)} - \dfrac{(x_2^{(k)})^2 - 2(x_1^{(k)})^2 + x_1^{(k)}x_2^{(k)} - 3x_2^{(k)} + 5}{x_2^{(k)} - 4x_1^{(k)}}, \\ x_2^{(k+1)} = x_2^{(k)} - \dfrac{(x_2^{(k)})^2 - 2(x_1^{(k)})^2 - 8x_1^{(k)}x_2^{(k)} + 12x_2^{(k)} - 5}{2(x_2^{(k)} - 4x_1^{(k)})}, \end{cases}$$

$$(k = 0,1,\cdots).$$

由 $x^{(0)} = (1.5, 1.0)^T$ 逐次迭代得到

$$x^{(1)} = (1.5, 0.75)^T,$$
$$x^{(2)} = (1.488095, 0.755952)^T,$$
$$x^{(3)} = (1.488034, 0.755983)^T.$$

$x^{(3)}$ 的每一位都是有效数字.

评　注

本章着重介绍求解单变量非线性方程 $f(x)=0$ 的迭代法及其理论. 不动点迭代、局部收敛性及收敛阶等基本概念是十分重要的,它很容易推广到非线性方程组. 在迭代法中以牛顿法最实用,它在单根附近具有 2 阶收敛,但应用时要选取较好的初始近似才能保证迭代收敛. 为克服这一缺点,可使用牛顿下山法. 斯蒂芬森方法可将一阶方法加速为二阶,也是值得重视的算法. 弦截法(或称割线法)与抛物线法(也称密勒法)是属于插值方法,它们不用算 $f(x)$ 的导数,又具有超线性收敛,也是常用的有效方法. 这类方法是多点迭代法,它不同于 $x_{k+1} = \varphi(x_k)$ 的单点迭代,计算时必须给出两个以上的初始近似. 其收敛性说明可参看文献[2]和[7].

非线性方程组的解法和理论是当今数值分析研究的重要课题之一,新方法不断出现,它也是科学计算经常遇到的. 这里我们只

简单介绍了牛顿法的迭代公式，有关理论均未提及，需要进一步了解的读者可参阅文献[1]和[17].

单个代数方程(即多项式方程)求根有久远的历史，也有不少特殊方法. 本章均未介绍，有关理论和算法读者可参阅文献[7]和[18]及其他文献.

习 题

1. 用二分法求方程 $x^2-x-1=0$ 的正根，要求误差小于 0.05.

2. 为求方程 $x^3-x^2-1=0$ 在 $x_0=1.5$ 附近的一个根，设将方程改写成下列等价形式，并建立相应的迭代公式.

(1) $x=1+1/x^2$，迭代公式 $x_{k+1}=1+1/x_k^2$；

(2) $x^3=1+x^2$，迭代公式 $x_{k+1}=\sqrt[3]{1+x_k^2}$；

(3) $x^2=\dfrac{1}{x-1}$，迭代公式 $x_{k+1}=1/\sqrt{x_k-1}$.

试分析每种迭代公式的收敛性，并选取一种公式求出具有四位有效数字的近似根.

3. 比较求 $e^x+10x-2=0$ 的根到三位小数所需的计算量：

(1) 在区间[0,1]内用二分法；

(2) 用迭代法 $x_{k+1}=(2-e^{x_k})/10$，取初值 $x_0=0$.

4. 给定函数 $f(x)$，设对一切 x，$f'(x)$存在且 $0<m\leqslant f'(x)\leqslant M$，证明对于范围 $0<\lambda<2/M$ 内的任意定数 λ，迭代过程 $x_{k+1}=x_k-\lambda f(x_k)$ 均收敛于 $f(x)=0$ 的根 x^*.

5. 用斯蒂芬森迭代法计算第 2 题中(2)，(3)的近似根，精确到 10^{-5}.

6. 设 $\varphi(x)=x-p(x)f(x)-q(x)f^2(x)$，试确定函数 $p(x)$和 $q(x)$，使求解 $f(x)=0$ 且以 $\varphi(x)$为迭代函数的迭代法至少三阶收敛.

7. 用下列方法求 $f(x)=x^3-3x-1=0$ 在 $x_0=2$ 附近的根. 根的准确值 $x^*=1.87938524\cdots$，要求计算结果准确到四位有效数字.

(1) 用牛顿法；

(2) 用弦截法，取 $x_0=2$，$x_1=1.9$；

(3) 用抛物线法，取 $x_0=1, x_1=3, x_2=2$.

8. 分别用二分法和牛顿法求 $x-\tan x=0$ 的最小正根.

9. 研究求$\sqrt{a}$的牛顿公式

$$x_{k+1}=\frac{1}{2}\left(x_k+\frac{a}{x_k}\right),\ x_0>0.$$

证明对一切 $k=1,2,\cdots, x_k\geqslant\sqrt{a}$ 且序列 $x_1, x_2, \cdots$ 是递减的.

10. 对于 $f(x)=0$ 的牛顿公式 $x_{k+1}=x_k-f(x_k)/f'(x_k)$，证明

$$R_k=(x_k-x_{k-1})/(x_{k-1}-x_{k-2})^2$$

收敛到 $-f''(x^*)/[2f'(x^*)]$，这里 x^* 为 $f(x)=0$ 的根.

11. 用牛顿法和求重根迭代法(4.13)和(4.14)计算方程 $f(x)=\left(\sin x-\frac{x}{2}\right)^2=0$ 的一个近似根，准确到 10^{-5}，初始值 $x_0=\frac{\pi}{2}$.

12. 应用牛顿法于方程 $x^3-a=0$，导出求立方根$\sqrt[3]{a}$的迭代公式，并讨论其收敛性.

13. 应用牛顿法于方程 $f(x)=1-\frac{a}{x^2}=0$，导出求$\sqrt{a}$的迭代公式，并用此公式求 $\sqrt{115}$ 的值.

14. 应用牛顿法于方程 $f(x)=x^n-a=0$ 和 $f(x)=1-\frac{a}{x^n}=0$，分别导出求$\sqrt[n]{a}$的迭代公式，并求

$$\lim_{k\to\infty}(\sqrt[n]{a}-x_{k+1})/(\sqrt[n]{a}-x_k)^2.$$

15. 证明迭代公式

$$x_{k+1}=\frac{x_k(x_k^2+3a)}{3x_k^2+a}$$

是计算$\sqrt{a}$的三阶方法. 假定初值 x_0 充分靠近根 x^*，求

$$\lim_{x\to\infty}(\sqrt{a}-x_{k+1})/(\sqrt{a}-x_k)^3.$$

16. 用牛顿法解方程组

$$\begin{cases}x^2+y^2=4,\\ x^2-y^2=1.\end{cases}$$

取 $\boldsymbol{x}^0=(1.6, 1.2)^{\mathrm{T}}$.

第 8 章　矩阵特征值问题计算

8.1　引　　言

物理、力学和工程技术中的很多问题在数学上都归结为求矩阵的特征值问题. 例如,振动问题(大型桥梁或建筑物的振动、机械的振动、电磁振荡等),物理学中某些临界值的确定,这些问题都归结为下述数学问题.

定义 1　(1) 已知 $\boldsymbol{A}=(a_{ij})_{n\times n}$,则称

$$\varphi(\lambda)=\det(\lambda\boldsymbol{I}-\boldsymbol{A})=\det\begin{pmatrix}\lambda-a_{11} & -a_{12} & \cdots & -a_{1n}\\ -a_{21} & \lambda-a_{22} & \cdots & -a_{2n}\\ \vdots & \vdots & \ddots & \vdots\\ -a_{n1} & -a_{n2} & \cdots & \lambda-a_{nn}\end{pmatrix}$$

$$=\lambda^n-(a_{11}+a_{22}+\cdots+a_{nn})\lambda^{n-1}+(\text{次级}\leqslant n-2\text{ 的项})$$

为 $\boldsymbol{A}$ 的**特征多项式**.

$\boldsymbol{A}$ 的特征方程

$$\varphi(\lambda)=\det(\lambda\boldsymbol{I}-\boldsymbol{A})=0 \tag{1.1}$$

一般有 n 个根(实的或复的,重根按重数计算)(当 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 时,$\varphi(\lambda)=0$为实系数 n 次代数方程,其复根共轭成对出现),称为 $\boldsymbol{A}$ 的**特征值**. 用 $\lambda(\boldsymbol{A})$表示 $\boldsymbol{A}$ 的所有特征值的**集合**.

(2) 设 λ 为 $\boldsymbol{A}$ 特征值,相应的齐次方程组

$$(\lambda\boldsymbol{I}-\boldsymbol{A})\boldsymbol{x}=\boldsymbol{0} \tag{1.2}$$

的非零解 $\boldsymbol{x}$ 称为矩阵 $\boldsymbol{A}$ 的对应于 λ 的**特征向量**.

例 1　求 $\boldsymbol{A}$ 的特征值及特征向量,其中

$$A = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 3 & 1 \\ 0 & 1 & 2 \end{pmatrix}.$$

解 矩阵 A 的特征方程为

$$\varphi(\lambda) = \det(\lambda I - A) = \lambda^3 - 7\lambda^2 + 14\lambda - 8 = 0,$$

求得 A 特征值为:

$$\lambda_1 = 1,\ \lambda_2 = 2,\ \lambda_3 = 4.$$

对应于各特征值矩阵 A 特征向量分别为:

$$x_1 = \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix},\ x_2 = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix},\ x_3 = \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}.$$

下面叙述有关特征值的一些结果.

定理 1 设 λ 为 $A \in \mathbf{R}^{n\times n}$ 的特征值且 $Ax = \lambda x$,其中 $x \neq 0$,则

(1) $c\lambda$ 为 cA 的特征值(c 为常数 $c \neq 0$);

(2) $\lambda - p$ 为 $A - pI$ 的特征值,即 $(A - pI)x = (\lambda - p)x$;

(3) λ^k 为 A^k 的特征值;

(4) 设 A 为非奇异阵,那么 $\lambda \neq 0$ 且 $\frac{1}{\lambda}$ 为 A^{-1} 特征值,即 $A^{-1}x = \frac{1}{\lambda}x$.

定理 2 设 $\lambda_i\,(i = 1, 2, \cdots, n)$ 为 n 阶矩阵 $A = (a_{ij})_{n\times n}$ 特征值,则

(1) $\sum_{i=1}^{n}\lambda_i = \sum_{i=1}^{n}a_{ii} = \mathrm{tr}(A)$;

(2) $\det(A) = \lambda_1\lambda_2\cdots\lambda_n$.

定理 3 设 $A \in \mathbf{R}^{n\times n}$,则

$$\lambda(A^{\mathrm{T}}) = \lambda(A).$$

定理 4 设 A 为分块上三角阵,即

$$A = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ & A_{22} & \cdots & A_{2m} \\ & & \ddots & \vdots \\ & & & A_{mm} \end{pmatrix},$$

其中每个对角块 A_{ii} 均为方阵，则 $\lambda(A) = \bigcup_{i=1}^{m} \lambda(A_{ii})$.

定理 5 设 A 与 B 为相似矩阵（即存在非奇异阵 P 使 $B=P^{-1}AP$），则

(1) A 与 B 有相同的特征值；

(2) 如果 y 是 B 特征向量，则 Py 是 A 特征向量.

定理 5 说明，一个矩阵 A 经过相似变换（$A \to B = P^{-1}AP$），则 A 的特征值不变.

定义 2 设 $A \in \mathbf{R}^{n\times n}$，如果 A 有一个重数为 k 的特征值 λ 且对应于 λ 矩阵 A 的线性无关的特征向量个数少于 k（一般 $\leqslant k$），称 A 为**亏损矩阵**.

一个亏损矩阵是一个没有足够特征向量的矩阵，亏损矩阵在理论上和计算上都存在困难.

定理 6 (1) $A \in \mathbf{R}^{n\times n}$ 可对角化，即存在非奇异矩阵 P 使

$$P^{-1}AP = \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{pmatrix}$$

的充要条件是 A 具有 n 个线性无关的特征向量.

(2) 如果 $A \in \mathbf{R}^{n\times n}$ 有 m 个（$m \leqslant n$）不同的特征值 $\lambda_1, \lambda_2, \cdots, \lambda_m$，则对应的特征向量 $x_1, x_2, \cdots, x_m$ 线性无关.

定理 7（对称矩阵的正交约化） 设 $A \in \mathbf{R}^{n\times n}$ 为对称矩阵，则：

(1) A 的特征值均为实数；

(2) $\boldsymbol{A}$ 有 n 个线性无关的特征向量；

(3) 存在一个正交矩阵 $\boldsymbol{P}$ 使得

$$\boldsymbol{P}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{P} = \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{pmatrix},$$

且 $\lambda_i(i=1,\cdots,n)$ 为 $\boldsymbol{A}$ 特征值，而 $\boldsymbol{P}=(\boldsymbol{u}_1,\boldsymbol{u}_2,\cdots,\boldsymbol{u}_n)$ 的列向量 $\boldsymbol{u}_j$ 为 $\boldsymbol{A}$ 的对应于 λ_j 的特征向量.

下面讨论矩阵特征值界的估计.

定义 3 设 $\boldsymbol{A}=(a_{ij})_{n\times n}$. 令：(1) $r_i=\sum\limits_{\substack{j=1\\j\neq i}}^{n}|a_{ij}|\quad(i=1,2,\cdots,n)$；(2) 集合 $D_i=\{z\mid |z-a_{ii}|\leqslant r_i,\ z\in\mathbf{C}\}$. 称复平面上以 a_{ii} 为圆心，以 r_i 为半径的所有圆盘为 $\boldsymbol{A}$ 的 Gerschgorin 圆盘.

定理 8(Gerschgorin 圆盘定理) (1) 设 $\boldsymbol{A}=(a_{ij})_{n\times n}$，则 $\boldsymbol{A}$ 的每一个特征值必属于下述某个圆盘之中

$$|\lambda-a_{ii}|\leqslant r_i=\sum_{\substack{j=1\\j\neq i}}^{n}|a_{ij}|\qquad(i=1,2,\cdots,n).$$

或者说，$\boldsymbol{A}$ 的特征值都在复平面上 n 个圆盘的并集中.

(2) 如果 $\boldsymbol{A}$ 有 m 个圆盘组成一个连通的并集 S，且 S 与余下 $n-m$ 个圆盘是分离的，则 S 内恰包含 $\boldsymbol{A}$ 的 m 个特征值.

特别地，如果 $\boldsymbol{A}$ 的一个圆盘 D_i 是与其他圆盘分离的(即孤立圆盘)，则 D_i 中精确地包含 $\boldsymbol{A}$ 的一个特征值.

证明 只就(1)给出证明. 设 λ 为 $\boldsymbol{A}$ 的特征值，即

$$\boldsymbol{A}\boldsymbol{x}=\lambda\boldsymbol{x}，其中\ \boldsymbol{x}=(x_1,x_2,\cdots,x_n)^{\mathrm{T}}\neq\boldsymbol{0}.$$

记 $|x_k|=\max\limits_{1\leqslant i\leqslant n}|x_i|=\|\boldsymbol{x}\|_\infty\neq 0$ 考虑 $\boldsymbol{A}\boldsymbol{x}=\lambda\boldsymbol{x}$ 的第 k 个方程，即

$$\sum_{j=1}^{n} a_{kj} x_j = \lambda x_k ,$$

或

$$(\lambda - a_{kk}) x_k = \sum_{j \neq k} a_{kj} x_j ,$$

于是$|\lambda - a_{kk}| \, |x_k| \leqslant \sum_{j \neq k} |a_{kj}| \, |x_j| \leqslant |x_k| \sum_{j \neq k} |a_{kj}|$,

即

$$|\lambda - a_{kk}| \leqslant \sum_{j \neq k} |a_{kj}| = r_k .$$

这说明,$\boldsymbol{A}$ 的每一个特征值必位于 $\boldsymbol{A}$ 的一个圆盘中,并且相应的特征值 λ 一定位于第 k 个圆盘中(其中 k 是对应特征向量 $\boldsymbol{x}$ 绝对值最大的分量的下标).

利用相似矩阵性质,有时可以获得 $\boldsymbol{A}$ 的特征值进一步的估计,即适当选取非奇异对角阵

$$\boldsymbol{D}^{-1} = \begin{pmatrix} \alpha_1^{-1} & & & \\ & \alpha_2^{-1} & & \\ & & \ddots & \\ & & & \alpha_n^{-1} \end{pmatrix}$$

并做相似变换 $\boldsymbol{D}^{-1}\boldsymbol{A}\boldsymbol{D} = \left(\dfrac{a_{ij}\alpha_j}{\alpha_i}\right)_{n\times n}$. 适当选取 $\alpha_i (i=1,2,\cdots,n)$ 可使某些圆盘半径及连通性发生变化.

例 2　估计矩阵

$$\boldsymbol{A} = \begin{pmatrix} 4 & 1 & 0 \\ 1 & 0 & -1 \\ 1 & 1 & -4 \end{pmatrix}$$

特征值的范围.

解　$\boldsymbol{A}$ 的 3 个圆盘为

$$D_1: \quad |\lambda - 4| \leqslant 1,$$

$$D_2: \quad |\lambda| \leqslant 2,$$

$$D_3: \quad |\lambda + 4| \leqslant 2.$$

由定理 8,可知 $\boldsymbol{A}$ 的 3 个特征值位于 3 个圆盘的并集中,由于 D_1 是孤立圆盘,所以 D_1 内恰好包含 $\boldsymbol{A}$ 的一个特征值 λ_1(为实特征值),即

$$3 \leqslant \lambda_1 \leqslant 5$$

$\boldsymbol{A}$ 的其他两个特征值 λ_2, λ_3 包含在 D_2, D_3 的并集中.

现选取对角阵

$$\boldsymbol{D}^{-1} = \begin{pmatrix} 1 & & \\ & 1 & \\ & & 0.9 \end{pmatrix}$$

做相似变换

$$\boldsymbol{A} \to \boldsymbol{A}_1 = \boldsymbol{D}^{-1}\boldsymbol{A}\boldsymbol{D} = \begin{pmatrix} 4 & 1 & 0 \\ 1 & 0 & -\dfrac{10}{9} \\ 0.9 & 0.9 & -4 \end{pmatrix}.$$

$\boldsymbol{A}_1$ 的 3 个圆盘为

$$E_1: \quad |\lambda - 4| \leqslant 1,$$

$$E_2: \quad |\lambda| \leqslant \frac{19}{9},$$

$$E_3: \quad |\lambda + 4| \leqslant 1.8.$$

显然,3 个圆盘都是孤立圆盘,所以,每一个圆盘都包含 $\boldsymbol{A}$ 的一个特征值(为实特征值)且有估计

$$\begin{cases} 3 \leqslant \lambda_1 \leqslant 5, \\ -\dfrac{19}{9} \leqslant \lambda_2 \leqslant \dfrac{19}{9}, \\ -5.8 \leqslant \lambda_3 \leqslant -2.2. \end{cases}$$

下面给出理论上有关通过酉相似变换及正交相似变换可以约化一般矩阵 $\boldsymbol{A} \in \mathbf{R}^{n\times n}$ 到什么程度的问题.

定理 9(Schur 定理) 设 $\boldsymbol{A} \in \mathbf{R}^{n\times n}$,则存在酉阵 $\boldsymbol{U}$ 使

$$U^{H}AU=\begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ & r_{22} & \cdots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \end{pmatrix}=\boldsymbol{R}\ (\text{上三角阵}),$$

其中 $r_{ii}(i=1,2,\cdots,n)$ 为 $\boldsymbol{A}$ 的特征值.

当 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 时，如果限制用正交相似变换，由于 $\boldsymbol{A}$ 有复的特征值，$\boldsymbol{A}$ 不能用正交相似变换约化为上三角阵. 用正交相似变换能将 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 约化到什么程度呢？

定理 10（实 Schur 分解）　设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$，则存在正交矩阵 $\boldsymbol{Q}$ 使

$$Q^{T}AQ=\begin{pmatrix} \boldsymbol{R}_{11} & \boldsymbol{R}_{12} & \cdots & \boldsymbol{R}_{1m} \\ & \boldsymbol{R}_{22} & \cdots & \boldsymbol{R}_{2m} \\ & & \ddots & \vdots \\ & & & \boldsymbol{R}_{mm} \end{pmatrix},$$

其中对角块 $\boldsymbol{R}_{ii}(i=1,2,\cdots,m)$ 为一阶或二阶方阵，且每个一阶 $\boldsymbol{R}_{ii}$ 是 $\boldsymbol{A}$ 的实特征值，每个二阶对角块 $\boldsymbol{R}_{jj}$ 的两个特征值是 $\boldsymbol{A}$ 的两个共轭复特征值.

我们转向实 Schur 型的实际计算.

定义 4　设 $\boldsymbol{A}$ 为 n 阶实对称矩阵，对于任一非零向量 $\boldsymbol{x}$，称

$$R(\boldsymbol{x})=\frac{(\boldsymbol{Ax},\boldsymbol{x})}{(\boldsymbol{x},\boldsymbol{x})}$$

为对应于向量 $\boldsymbol{x}$ 的**瑞利(Rayleigh)商**.

定理 11　设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 为对称矩阵（其特征值次序记为 $\lambda_1\geqslant\lambda_2\geqslant\cdots\geqslant\lambda_n$），则

1. $\lambda_n\leqslant\dfrac{(\boldsymbol{Ax},\boldsymbol{x})}{(\boldsymbol{x},\boldsymbol{x})}\leqslant\lambda_1$　　（对任何非零 $\boldsymbol{x}\in\mathbf{R}^n$）；

2. $\lambda_1=\max\limits_{\substack{\boldsymbol{x}\in\mathbf{R}^n\\ \boldsymbol{x}\neq\boldsymbol{0}}}\dfrac{(\boldsymbol{Ax},\boldsymbol{x})}{(\boldsymbol{x},\boldsymbol{x})}$；

3. $\lambda_n=\min\limits_{\substack{x\in\mathbf{R}^n\\x\neq 0}}\frac{(\boldsymbol{A}\boldsymbol{x},\boldsymbol{x})}{(\boldsymbol{x},\boldsymbol{x})}$.

证明 只证 1，关于 2，3 留作习题.

由于 $\boldsymbol{A}$ 为实对称矩阵，可将 $\lambda_1,\lambda_2,\cdots,\lambda_n$ 对应的特征向量 $\boldsymbol{x}_1$，$\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n$ 正交规范化，则有 $(\boldsymbol{x}_i,\boldsymbol{x}_j)=\delta_{ij}$. 设 $\boldsymbol{x}\neq\boldsymbol{0}$ 为 $\mathbf{R}^n$ 中任一向量，则有展开式

$$\boldsymbol{x}=\sum_{i=1}^{n}\alpha_i\boldsymbol{x}_i,\quad \|\boldsymbol{x}\|_2=\left(\sum_{i=1}^{n}\alpha_i^2\right)^2\neq 0,$$

于是

$$\frac{(\boldsymbol{A}\boldsymbol{x},\boldsymbol{x})}{(\boldsymbol{x},\boldsymbol{x})}=\frac{\sum\limits_{i=1}^{n}\alpha_i^2\lambda_i}{\sum\limits_{i=1}^{n}\alpha_i^2}.$$

从而 1 成立. 结论 1 说明瑞利商必位于 λ_n 和 λ_1 之间.

关于计算矩阵 $\boldsymbol{A}$ 的特征值问题，当 $n=2,3$ 时，我们还可按行列式展开的办法求 $\varphi(\lambda)=0$ 的根. 但当 n 较大时，如果按展开行列式的办法，首先求出 $\varphi(\lambda)$ 的系数，再求 $\varphi(\lambda)$ 的根，工作量就非常大，用这种办法求矩阵特征值是不切实际的，由此需要研究求 $\boldsymbol{A}$ 的特征值及特征向量的数值解法.

本章将介绍一些计算机上常用的两类方法，一类是幂法及反幂法(迭代法)，另一类是正交相似变换的方法(变换法).

8.2 幂法及反幂法

8.2.1 幂法

幂法是一种计算矩阵主特征值(矩阵按模最大的特征值)及对应特征向量的迭代方法，特别适用于大型稀疏矩阵. 反逆法是计算海森伯格阵或三对角阵的对应一个给定近似特征值的特征向量的

有效方法之一.

设实矩阵 $\boldsymbol{A}=(a_{ij})_{n\times n}$有一个完全的特征向量组,其特征值为 $\lambda_1,\lambda_2,\cdots,\lambda_n$,相应的特征向量为 $\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n$. 已知 $\boldsymbol{A}$ 的主特征值是实根,且满足条件

$$|\lambda_1|>|\lambda_2|\geqslant|\lambda_3|\geqslant\cdots\geqslant|\lambda_n|, \tag{2.1}$$

现讨论求 λ_1 及 $\boldsymbol{x}_1$ 的方法.

幂法的基本思想是任取一个非零的初始向量 $\boldsymbol{v}_0$,由矩阵 $\boldsymbol{A}$ 构造一向量序列

$$\begin{cases} \boldsymbol{v}_1=\boldsymbol{A}\boldsymbol{v}_0 \\ \boldsymbol{v}_2=\boldsymbol{A}\boldsymbol{v}_1=\boldsymbol{A}^2\boldsymbol{v}_0, \\ \qquad\cdots \\ \boldsymbol{v}_{k+1}=\boldsymbol{A}\boldsymbol{v}_k=\boldsymbol{A}^{k+1}\boldsymbol{v}_0, \\ \qquad\cdots \end{cases} \tag{2.2}$$

称为迭代向量. 由假设,$\boldsymbol{v}_0$ 可表示为

$$\boldsymbol{v}_0=\alpha_1\boldsymbol{x}_1+\alpha_2\boldsymbol{x}_2+\cdots+\alpha_n\boldsymbol{x}_n \qquad (设\ \alpha_1\neq 0), \tag{2.3}$$

于是

$$\begin{aligned}\boldsymbol{v}_k&=\boldsymbol{A}\boldsymbol{v}_{k-1}=\boldsymbol{A}^k\boldsymbol{v}_0=\alpha_1\lambda_1^k\boldsymbol{x}_1+\alpha_2\lambda_2^k\boldsymbol{x}_2+\cdots+\alpha_n\lambda_n^k\boldsymbol{x}_n \\ &=\lambda_1^k\left[\alpha_1\boldsymbol{x}_1+\sum_{i=2}^n\alpha_i(\lambda_i/\lambda_1)^k\boldsymbol{x}_i\right]\equiv\lambda_1^k(\alpha_1\boldsymbol{x}_1+\boldsymbol{\varepsilon}_k),\end{aligned}$$

其中 $\boldsymbol{\varepsilon}_k=\sum_{i=2}^n\alpha_i(\lambda_i/\lambda_1)^k\boldsymbol{x}_i$. 由假设

$|\lambda_i/\lambda_1|<1(i=2,3,\cdots,n)$,故$\lim_{k\to\infty}\boldsymbol{\varepsilon}_k=\boldsymbol{0}$, 从而

$$\lim_{k\to\infty}\frac{\boldsymbol{v}_k}{\lambda_1^k}=\alpha_1\boldsymbol{x}_1. \tag{2.4}$$

这说明序列$\frac{\boldsymbol{v}_k}{\lambda_1^k}$越来越接近 $\boldsymbol{A}$ 的对应于λ_1 的特征向量,或者说当 k 充分大时

$$\boldsymbol{v}_k\approx\alpha_1\lambda_1^k\boldsymbol{x}_1, \tag{2.5}$$

即迭代向量 $\boldsymbol{v}_k$ 为 λ_1 的特征向量的近似向量(除一个因子外).

下面再考虑主特征值 λ_1 的计算,用 $(\boldsymbol{v}_k)_i$ 表示 $\boldsymbol{v}_k$ 的第 i 个分量,则

$$\frac{(\boldsymbol{v}_{k+1})_i}{(\boldsymbol{v}_k)_i} = \lambda_1 \left\{ \frac{\alpha_1 (\boldsymbol{x}_1)_i + (\boldsymbol{\varepsilon}_{k+1})_i}{\alpha_1 (\boldsymbol{x}_1)_i + (\boldsymbol{\varepsilon}_k)_i} \right\}, \tag{2.6}$$

故

$$\lim_{k\to\infty} \frac{(\boldsymbol{v}_{k+1})_i}{(\boldsymbol{v}_k)_i} = \lambda_1, \tag{2.7}$$

也就是说两相邻迭代向量分量的比值收敛到主特征值.

这种由已知非零向量 $\boldsymbol{v}_0$ 及矩阵 $\boldsymbol{A}$ 的乘幂 $\boldsymbol{A}^k$ 构造向量序列 $\{\boldsymbol{v}_k\}$ 以计算 $\boldsymbol{A}$ 的主特征值 λ_1(利用(2.7)式)及相应特征向量(利用(2.5)式)的方法称为**幂法**.

由(2.6)式知,$\dfrac{(\boldsymbol{v}_{k+1})_i}{(\boldsymbol{v}_k)_i} \to \lambda_1$ 的收敛速度由比值 $r = \left|\dfrac{\lambda_2}{\lambda_1}\right|$ 来确定,r 越小收敛越快,但当 $r = \left|\dfrac{\lambda_2}{\lambda_1}\right| \approx 1$ 时收敛可能就很慢.

总结上述讨论,有

定理 12 设 $\boldsymbol{A} \in \mathbf{R}^{n\times n}$ 有 n 个线性无关的特征向量,主特征值 λ_1 满足

$$|\lambda_1| > |\lambda_2| \geqslant |\lambda_3| \geqslant \cdots \geqslant |\lambda_n|,$$

则对任何非零初始向量 $\boldsymbol{v}(\alpha_1 \neq 0)$,(2.4),(2.7)式成立.

如果 $\boldsymbol{A}$ 的主特征值为实的重根,即 $\lambda_1 = \lambda_2 = \cdots = \lambda_r$,且

$$|\lambda_r| > |\lambda_{r+1}| \geqslant \cdots \geqslant |\lambda_n|,$$

又设 $\boldsymbol{A}$ 有 n 个线性无关的特征向量,λ_1 对应的 r 个线性无关特征向量为 $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_r$,则由(2.2)式

$$\boldsymbol{v}_k = \boldsymbol{A}^k \boldsymbol{v}_0 = \lambda_1^k \left\{ \sum_{i=1}^{r} \alpha_i \boldsymbol{x}_i + \sum_{i=r+1}^{n} \alpha_i (\lambda_i/\lambda_1)^k \boldsymbol{x}_i \right\},$$

$$\lim_{k\to\infty} \frac{\boldsymbol{v}_k}{\lambda_1^k} = \sum_{i=1}^{r} \alpha_i \boldsymbol{x}_i \qquad \left(设 \sum_{i=1}^{r} \alpha_i \boldsymbol{x}_i \neq \boldsymbol{0}\right).$$

这说明当 $\boldsymbol{A}$ 的主特征值是实的重根时，定理 5 的结论还是正确的.

应用幂法计算 $\boldsymbol{A}$ 的主特征值 λ_1 及对应的特征向量时，如果 $|\lambda_1|>1$（或 $|\lambda_1|<1$），迭代向量 $\boldsymbol{v}_k$ 的各个不等于零的分量将随 $k\to\infty$ 而趋向于无穷（或趋于零），这样在计算机实现时就可能"溢出". 为了克服这个缺点，就需要将迭代向量加以规范化.

设有一向量 $\boldsymbol{v}\neq\boldsymbol{0}$，将其规范化得到向量

$$\boldsymbol{u}=\frac{\boldsymbol{v}}{\max(\boldsymbol{v})},$$

其中 $\max(\boldsymbol{v})$ 表示向量 $\boldsymbol{v}$ 的绝对值最大的分量，即如果有

$$|v_{i_0}|=\max_{1\leqslant i\leqslant n}|v_i|,$$

则 $\max(\boldsymbol{v})=v_{i_0}$，且 i_0 为所有绝对值最大的分量中的最小下标.

在定理 12 的条件下幂法可这样进行：任取一初始向量 $\boldsymbol{v}_0\neq\boldsymbol{0}$ $(\alpha_1\neq 0)$，构造向量序列 $\max(\boldsymbol{v})$

$$\begin{cases}\boldsymbol{v}_1=\boldsymbol{A}\boldsymbol{u}_0=\boldsymbol{A}\boldsymbol{v}_0, & \boldsymbol{u}_1=\dfrac{\boldsymbol{v}_1}{\max(\boldsymbol{v}_1)}=\dfrac{\boldsymbol{A}\boldsymbol{v}_0}{\max(\boldsymbol{A}\boldsymbol{v}_0)},\\ \boldsymbol{v}_2=\boldsymbol{A}\boldsymbol{u}_1=\dfrac{\boldsymbol{A}^2\boldsymbol{v}_0}{\max(\boldsymbol{A}\boldsymbol{v}_0)}, & \boldsymbol{u}_2=\dfrac{\boldsymbol{v}_2}{\max(\boldsymbol{v}_2)}=\dfrac{\boldsymbol{A}^2\boldsymbol{v}_0}{\max(\boldsymbol{A}^2\boldsymbol{v}_0)},\\ \quad\cdots & \quad\cdots\\ \boldsymbol{v}_k=\dfrac{\boldsymbol{A}^k\boldsymbol{v}_0}{\max(\boldsymbol{A}^{k-1}\boldsymbol{v}_0)}, & \boldsymbol{u}_k=\dfrac{\boldsymbol{A}^k\boldsymbol{v}_0}{\max(\boldsymbol{A}^k\boldsymbol{v}_0)},\end{cases}$$

由(2.3)式

$$\boldsymbol{A}^k\boldsymbol{v}_0=\sum_{i=1}^n\alpha_i\lambda_i^k\boldsymbol{x}_i=\lambda_i^k\left[\alpha_1\boldsymbol{x}_1+\sum_{i=2}^n\alpha_i\left(\frac{\lambda_i}{\lambda_1}\right)^k\boldsymbol{x}_i\right],\qquad(2.8)$$

$$\boldsymbol{u}_k=\frac{\boldsymbol{A}^k\boldsymbol{v}_0}{\max(\boldsymbol{A}^k\boldsymbol{v}_0)}=\frac{\lambda_1^k\left[\alpha_1\boldsymbol{x}_1+\sum_{i=2}^n\alpha_i\left(\frac{\lambda_i}{\lambda_1}\right)^k\boldsymbol{x}_i\right]}{\max\left[\lambda_1^k\left(\alpha_1\boldsymbol{x}_1+\sum_{i=2}^n\alpha_i\left(\frac{\lambda_i}{\lambda_1}\right)^k\boldsymbol{x}_i\right)\right]}$$

$$=\frac{\alpha_1\boldsymbol{x}_1+\sum_{i=2}^{n}\alpha_i\left(\frac{\lambda_i}{\lambda_1}\right)^k\boldsymbol{x}_i}{\max\left[\alpha_1\boldsymbol{x}_1+\sum_{i=2}^{n}\alpha_i\left(\frac{\lambda_i}{\lambda_1}\right)^k\boldsymbol{x}_i\right]}\to\frac{\boldsymbol{x}_1}{\max(\boldsymbol{x}_1)}(k\to\infty).$$

这说明规范化向量序列收敛到主特征值对应的特征向量.

同理,可得到

$$\boldsymbol{v}_k=\frac{\lambda_1^k\left[\alpha_1\boldsymbol{x}_1+\sum_{i=2}^{n}\alpha_i\left(\frac{\lambda_i}{\lambda_1}\right)^k\boldsymbol{x}_i\right]}{\max\left[\lambda_1^{k-1}\alpha_1\boldsymbol{x}_1+\sum_{i=2}^{n}\alpha_i\left(\frac{\lambda_i}{\lambda_1}\right)^{k-1}\boldsymbol{x}_i\right]},$$

$$\max(\boldsymbol{v}_k)=\frac{\lambda_1\max\left[\alpha_1\boldsymbol{x}_1+\sum_{i=2}^{n}\alpha_i\left(\frac{\lambda_i}{\lambda_1}\right)^k\boldsymbol{x}_i\right]}{\max\left[\alpha_1\boldsymbol{x}_1+\sum_{i=2}^{n}\alpha_i\left(\frac{\lambda_i}{\lambda_1}\right)^{k-1}\boldsymbol{x}_i\right]}\to\lambda_1,$$

$$(k\to\infty),$$

收敛速度由比值 $r=|\lambda_2/\lambda_1|$ 确定.总结上述讨论,有

定理 13 设 $\mathbf{A}\in\mathbf{R}^{n\times n}$ 有 n 个线性无关的特征向量,主特征值 λ_1 满足 $|\lambda_1|>|\lambda_2|\geqslant|\lambda_3|\geqslant\cdots\geqslant|\lambda_n|$,则对任意非零初始向量 $\boldsymbol{v}_0=\boldsymbol{u}_0(\alpha_1\neq0)$,按下述方法构造的向量序列 $\{\boldsymbol{u}_k\},\{\boldsymbol{v}_k\}$:

$$\begin{cases}\boldsymbol{v}_0=\boldsymbol{u}_0\neq\boldsymbol{0},\\ \boldsymbol{v}_k=\boldsymbol{A}\boldsymbol{u}_{k-1},\\ \mu_k=\max(\boldsymbol{v}_k),\\ \boldsymbol{u}_k=\boldsymbol{v}_k/\mu_k.\end{cases}\quad(k=1,2,\cdots),\tag{2.9}$$

则有

(1) $\lim\limits_{k\to\infty}\boldsymbol{u}_n=\dfrac{\boldsymbol{x}_1}{\max(\boldsymbol{x}_1)}$,

(2) $\lim\limits_{k\to\infty}\mu_k=\lambda_1$.

例 3 用幂法计算

$$A = \begin{pmatrix} 1.0 & 1.0 & 0.5 \\ 1.0 & 1.0 & 0.25 \\ 0.5 & 0.25 & 2.0 \end{pmatrix}$$

的主特征值和相应的特征向量. 计算过程如表 8-1.

下述结果是用 8 位浮点数字进行运算得到的,$\boldsymbol{u}_k$ 的分量值是舍入值. 于是得到

$$\lambda_1 \approx 2.5365323$$

及相应的特征向量$(0.7482, 0.6497, 1)^{\mathrm{T}}$. λ_1 和相应的特征向量的真值(8 位数字)为

$$\lambda_1 = 2.5365258,$$
$$\tilde{\boldsymbol{x}}_1 = (0.74822116, 0.64966116, 1)^{\mathrm{T}}.$$

表　8-1

k	$\boldsymbol{u}_k^{\mathrm{T}}$(规范化向量)			$\max(\boldsymbol{v}_k)$
0	(1	1	1)	
1	(0.9091	0.8182	1)	2.7500000
5	(0.7651	0.6674	1)	2.5587918
10	(0.7494	0.6508	1)	2.5380029
15	(0.7483	0.6497	1)	2.5366256
16	(0.7483	0.6497	1)	2.5365840
17	(0.7482	0.6497	1)	2.5365598
18	(0.7482	0.6497	1)	2.5365456
19	(0.7482	0.6497	1)	2.5365374
20	(0.7482	0.6497	1)	2.5365323

8.2.2　加速方法

原点平移法

由前面讨论知道,应用幂法计算 $\boldsymbol{A}$ 的主特征值的收敛速度主

要由比值 $r=\frac{\lambda_1}{\lambda_2}$ 来决定，但当 r 接近于 1 时，收敛可能很慢. 这时，一个补救的办法是采用加速收敛的方法.

引进矩阵

$$\boldsymbol{B}=\boldsymbol{A}-p\boldsymbol{I},$$

其中 p 为选择参数. 设 $\boldsymbol{A}$ 的特征值为 $\lambda_1,\lambda_2,\cdots,\lambda_n$，则 $\boldsymbol{B}$ 的相应特征值为 $\lambda_1-p,\lambda_2-p,\cdots,\lambda_n-p$，而且 $\boldsymbol{A},\boldsymbol{B}$ 的特征向量相同.

如果需要计算 $\boldsymbol{A}$ 的主特征值 λ_1，就要适当选择 p 使 λ_1-p 仍然是 $\boldsymbol{B}$ 的主特征值，且使

$$\left|\frac{\lambda_2-p}{\lambda_1-p}\right|<\left|\frac{\lambda_2}{\lambda_1}\right|.$$

对 $\boldsymbol{B}$ 应用幂法，使得在计算 $\boldsymbol{B}$ 的主特征值 λ_1-p 的过程中得到加速. 这种方法通常称为原点平移法. 对于 $\boldsymbol{A}$ 的特征值的某种分布，它是十分有效的.

例 4 设 $\boldsymbol{A}\in\mathbf{R}^{4\times4}$ 有特征值

$$\lambda_j=15-j\quad(j=1,2,3,4),$$

比值 $r=\frac{\lambda_2}{\lambda_1}\approx0.9$. 作变换

$$\boldsymbol{B}=\boldsymbol{A}-p\boldsymbol{I}\quad(p=12),$$

则 $\boldsymbol{B}$ 的特征值为

$$\mu_1=2,\mu_2=1,\mu_3=0,\mu_4=-1.$$

应用幂法计算 $\boldsymbol{B}$ 的主特征值 μ_1 的收敛速度的比值为

$$\left|\frac{\mu_2}{\mu_1}\right|=\left|\frac{\lambda_2-p}{\lambda_1-p}\right|=\frac{1}{2}<\left|\frac{\lambda_2}{\lambda_1}\right|\approx0.9.$$

虽然常常能够选择有利的 p 值，使幂法得到加速，但设计一个自动选择适当参数 p 的过程是困难的.

下面考虑当 $\boldsymbol{A}$ 的特征值是实数时，怎样选择 p 使采用幂法计

算 λ_1 得到加速.

设 **A** 的特征值满足

$$\lambda_1 > \lambda_2 \geqslant \cdots \geqslant \lambda_{n-1} > \lambda_n, \tag{2.10}$$

则不管 p 如何,$\boldsymbol{B}=\boldsymbol{A}-p\boldsymbol{I}$ 的主特征值为 λ_1-p 或 λ_n-p. 当我们希望计算 λ_1 及 $\boldsymbol{x}_1$ 时,首先应选择 p 使

$$|\lambda_1-p|>|\lambda_n-p|,$$

且使收敛速度的比值

$$\omega=\max\left\{\frac{|\lambda_2-p|}{|\lambda_1-p|},\frac{|\lambda_n-p|}{|\lambda_1-p|}\right\}=\min.$$

显然,当$\dfrac{\lambda_2-p}{\lambda_1-p}=-\dfrac{\lambda_n-p}{\lambda_1-p}$,即 $p=\dfrac{\lambda_2+\lambda_n}{2}\equiv p^*$ 时 ω 为最小,这时收敛速度的比值为

$$\frac{\lambda_2-p^*}{\lambda_1-p^*}=-\frac{\lambda_n-p^*}{\lambda_1-p^*}\equiv\frac{\lambda_2-\lambda_n}{2\lambda_1-\lambda_2-\lambda_n}.$$

当 **A** 的特征值满足(2.10)且 λ_2,λ_n 能初步估计时,我们就能确定 p^* 的近似值.

当希望计算 λ_n 时,应选择

$$p=\frac{\lambda_1+\lambda_{n-1}}{2}=p^*,$$

使得应用幂法计算 λ_n 得到加速.

例 5　计算例 3 中矩阵 **A** 的主特征值.

作变换 $\boldsymbol{B}=\boldsymbol{A}-p\boldsymbol{I}$,取 $p=0.75$,则

$$\boldsymbol{B}=\begin{pmatrix}0.25 & 1 & 0.5\\ 1 & 0.25 & 0.25\\ 0.5 & 0.25 & 1.25\end{pmatrix}.$$

对 **B** 应用幂法,计算结果如表 8-2.

表 8-2

k	u_k^{T}(规范化向量)			$\max(v_k)$
0	(1	1	1)	
5	(0.7516	0.6522	1)	1.7914011
6	(0.7491	0.6511	1)	1.7888443
7	(0.7488	0.6501	1)	1.7873300
8	(0.7484	0.6499	1)	1.7869152
9	(0.7483	0.6497	1)	1.7866587
10	(0.7482	0.6497	1)	1.7865914

由此得 $\boldsymbol{B}$ 的主特征值为 $\mu_1 \approx 1.7865914$,$\boldsymbol{A}$ 的主特征值 λ_1 为

$$\lambda_1 \approx \mu_1 + 0.75 = 2.5365914,$$

与例 3 结果比较,上述结果比例 3 迭代 15 次还好.若迭代 15 次,$\mu_1 = 1.7865258$(相应的 $\lambda_1 = 2.5365258$).

原点位移的加速方法,是一个矩阵变换方法.这种变换容易计算,又不破坏矩阵 $\boldsymbol{A}$ 的稀疏性,但 p 的选择依赖于对 $\boldsymbol{A}$ 的特征值分布的大致了解.

瑞利商加速

由定理 11 知,对称矩阵 $\boldsymbol{A}$ 的 λ_1 及 λ_n 可用瑞利商的极值来表示.下面我们将把瑞利商应用到用幂法计算实对称矩阵 $\boldsymbol{A}$ 的主特征值的加速收敛上来.

定理 14 设 $\boldsymbol{A} \in \mathbf{R}^{n\times n}$ 为对称矩阵,特征值满足

$$|\lambda_1| > |\lambda_2| \geqslant |\lambda_3| \geqslant \cdots \geqslant |\lambda_n|,$$

对应的特征向量满足 $(\boldsymbol{x}_i, \boldsymbol{x}_j) = \delta_{ij}$,应用幂法(公式(2.9))计算 $\boldsymbol{A}$ 的主特征值 λ_1,则规范化向量 $\boldsymbol{u}_k$ 的瑞利商给出 λ_1 的较好的近似

$$\frac{(\boldsymbol{A}\boldsymbol{u}_k, \boldsymbol{u}_k)}{(\boldsymbol{u}_k, \boldsymbol{u}_k)} = \lambda_1 + O\left(\left(\frac{\lambda_2}{\lambda_1}\right)^{2k}\right).$$

证明 由(2.8)式及

$$\boldsymbol{u}_k=\frac{\boldsymbol{A}^k\boldsymbol{u}_0}{\max(\boldsymbol{A}^k\boldsymbol{u}_0)},\boldsymbol{v}_{k+1}=\boldsymbol{A}\boldsymbol{u}_k=\frac{\boldsymbol{A}^{k+1}\boldsymbol{u}_0}{\max(\boldsymbol{A}^k\boldsymbol{u}_0)},$$

得

$$\frac{(\boldsymbol{A}\boldsymbol{u}_k,\boldsymbol{u}_k)}{(\boldsymbol{u}_k,\boldsymbol{u}_k)}=\frac{(\boldsymbol{A}^{k+1}\boldsymbol{u}_0,\boldsymbol{A}^k\boldsymbol{u}_0)}{(\boldsymbol{A}^k\boldsymbol{u}_0,\boldsymbol{A}^k\boldsymbol{u}_0)}=\frac{\sum\limits_{j=1}\alpha_j^2\lambda_j^{2k+1}}{\sum\limits_{j=1}^{n}\alpha_j^2\lambda_j^{2k}}$$

$$=\lambda_1+O\left(\left(\frac{\lambda_2}{\lambda_1}\right)^{2k}\right). \tag{2.11}$$

8.2.3 反幂法

反幂法用来计算矩阵按模最小的特征值及其特征向量，也可用来计算对应于一个给定近似特征值的特征向量.

设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 为非奇异矩阵，$\boldsymbol{A}$ 的特征值次序记为

$$|\lambda_1|\geqslant|\lambda_2|\geqslant\cdots\geqslant|\lambda_n|,$$

相应的特征向量为 $\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n$，则 $\boldsymbol{A}^{-1}$ 的特征值为

$$\left|\frac{1}{\lambda_n}\right|\geqslant\left|\frac{1}{\lambda_{n-1}}\right|\geqslant\cdots\geqslant\left|\frac{1}{\lambda_1}\right|,$$

对应的特征向量为 $\boldsymbol{x}_n,\boldsymbol{x}_{n-1},\cdots,\boldsymbol{x}_1$.

因此计算 $\boldsymbol{A}$ 的按模最小的特征值 λ_n 的问题就是计算 $\boldsymbol{A}^{-1}$ 的按模最大的特征值的问题.

对于 $\boldsymbol{A}^{-1}$ 应用幂法迭代(称为反幂法)，可求得矩阵 $\boldsymbol{A}^{-1}$ 的主特征值 $1/\lambda_n$，从而求得 $\boldsymbol{A}$ 的按模最小的特征值 λ_n.

反幂法迭代公式为：

任取初始向量 $\boldsymbol{v}_0=\boldsymbol{u}_0\neq\mathbf{0}$，构造向量序列

$$\begin{cases}\boldsymbol{v}_k=\boldsymbol{A}^{-1}\boldsymbol{u}_{k-1}\\ \boldsymbol{u}_k=\dfrac{\boldsymbol{v}_k}{\max(\boldsymbol{v}_k)}\end{cases}\quad(k=1,2,\cdots).$$

迭代向量 v_k 可以通过解方程组

$$Av_k = u_{k-1}$$

求得.

定理 15 设 A 为非奇异矩阵且有 n 个线性无关的特征向量，其对应的特征值满足

$$|\lambda_1| \geqslant |\lambda_2| \geqslant \cdots \geqslant |\lambda_{n-1}| > |\lambda_n| > 0,$$

则对任何初始非零向量 u_0 $(\alpha_n \neq 0)$，由反幂法构造的向量序列 $\{v_k\}$，$\{u_k\}$满足

(1) $\lim\limits_{k\to\infty} u_k = \dfrac{x_n}{\max(x_k)}$,

(2) $\lim\limits_{k\to\infty} \max(v_k) = \dfrac{1}{\lambda_n}$.

收敛速度的比值为 $\left|\dfrac{\lambda_n}{\lambda_{n-1}}\right|$.

在反幂法中也可以用原点平移法来加速迭代过程或求其他特征值及特征向量.

如果矩阵$(A-pI)^{-1}$存在，显然其特征值为

$$\frac{1}{\lambda_1 - p}, \frac{1}{\lambda_2 - p}, \cdots, \frac{1}{\lambda_n - p},$$

对应的特征向量仍然是 $x_1, x_2, \cdots, x_n$. 现对矩阵$(A-pI)^{-1}$应用幂法，得到反幂法的迭代公式

$$\begin{cases} u_0 = v_0 \neq \mathbf{0}, \text{初始向量} \\ v_k = (A - pI)^{-1} u_{k-1} \\ u_k = \dfrac{v_k}{\max(v_k)} \end{cases} \quad (k = 1, 2, \cdots). \tag{2.12}$$

如果 p 是 A 的特征值 λ_j 的一个近似值，且设 λ_j 与其他特征值是分离的，即

$$|\lambda_j - p| \ll |\lambda_i - p| \quad (i \neq j),$$

就是说$\dfrac{1}{\lambda_j - p}$是$(A-pI)^{-1}$的主特征值，可用反幂法(2.12)计算特

征值及特征向量.

设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 有 n 个线性无关的特征向量 $\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n$，则

$$\boldsymbol{u}_0=\sum_{i=1}^{n}\alpha_i\boldsymbol{x}_i\quad(\alpha_j\neq 0),$$

$$\boldsymbol{v}_k=\frac{(\boldsymbol{A}-p\boldsymbol{I})^{-k}\boldsymbol{u}_0}{\max((\boldsymbol{A}-p\boldsymbol{I})^{-(k-1)}\boldsymbol{u}_0)},$$

$$\boldsymbol{u}_k=\frac{(\boldsymbol{A}-p\boldsymbol{I})^{-k}\boldsymbol{u}_0}{\max((\boldsymbol{A}-p\boldsymbol{I})^{-k}\boldsymbol{u}_0)},$$

其中

$$(\boldsymbol{A}-p\boldsymbol{I})^{-k}\boldsymbol{u}_0=\sum_{i=1}^{n}\alpha_i(\lambda_i-p)^{-k}\boldsymbol{x}_i,$$

同理可得：

定理 16 设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 有 n 个线性无关的特征向量，$\boldsymbol{A}$ 的特征值及对应的特征向量分别记为 λ_i 及 $\boldsymbol{x}_i(i=1,2,\cdots,n)$，而 p 为 λ_j 的近似值，$(\boldsymbol{A}-p\boldsymbol{I})^{-1}$ 存在，且

$$|\lambda_j-p|\ll|\lambda_i-p|\quad(i\neq j).$$

则对任意的非零初始向量 $\boldsymbol{u}_0\alpha_j\neq 0$，由反幂法迭代公式(2.12)构造的向量序列 $\{\boldsymbol{v}_k\},\{\boldsymbol{u}_k\}$ 满足

(1) $\lim\limits_{k\to\infty}\boldsymbol{u}_k=\dfrac{\boldsymbol{x}_j}{\max(\boldsymbol{x}_j)}$；

(2) $\lim\limits_{k\to\infty}\max(\boldsymbol{v}_k)=\dfrac{1}{\lambda_j-p}$，即

$$p+\frac{1}{\max(\boldsymbol{v}_k)}\to\lambda_j\quad(当\ k\to\infty),$$

且收敛速度由比值 $r=|\lambda_j-p|/\min\limits_{i\neq j}|\lambda_i-p|$ 确定.

由该定理知，对 $\boldsymbol{A}-p\boldsymbol{I}$(其中 $p\approx\lambda_j$)应用反幂法，可用来计算特征向量 $\boldsymbol{x}_j$. 只要选择的 p 是 λ_j 的一个较好的近似且特征值分离情况较好，一般 r 很小，常常只要迭代一二次就可完成特征向量的计算.

反幂法迭代公式中的 $\boldsymbol{v}_k$ 是通过解方程组

$$(\boldsymbol{A}-p\boldsymbol{I})\boldsymbol{v}_k=\boldsymbol{u}_{k-1}$$

求得的. 为了节省工作量,可以先将 $\boldsymbol{A}-p\boldsymbol{I}$ 进行三角分解

$$\boldsymbol{P}(\boldsymbol{A}-p\boldsymbol{I})=\boldsymbol{L}\boldsymbol{U},$$

其中 $\boldsymbol{P}$ 为某个排列阵,于是求 $\boldsymbol{v}_k$ 相当于解两个三角形方程组

$$\boldsymbol{L}\boldsymbol{y}_k=\boldsymbol{P}\boldsymbol{u}_{k-1},$$

$$\boldsymbol{U}\boldsymbol{v}_k=\boldsymbol{y}_k.$$

实验表明,按下述方法选择 $\boldsymbol{u}_0$ 是较好的:选 $\boldsymbol{u}_0$ 使

$$\boldsymbol{U}\boldsymbol{v}_1=\boldsymbol{L}^{-1}\boldsymbol{P}\boldsymbol{u}_0=(1,1,\cdots,1)^{\mathrm{T}} \tag{2.13}$$

用回代求解(2.13)即得 $\boldsymbol{v}_1$,然后再按公式(2.12)进行迭代.

反幂法计算公式

1. 分解计算

$$\boldsymbol{P}(\boldsymbol{A}-p\boldsymbol{I})=\boldsymbol{L}\boldsymbol{U},\text{且保存 }\boldsymbol{L},\boldsymbol{U}\text{ 及 }\boldsymbol{P}\text{ 信息.}$$

2. 反幂法迭代

(1) 解 $\boldsymbol{U}\boldsymbol{v}_1=(1,\cdots,1)^{\mathrm{T}}$ 求 $\boldsymbol{v}_1$

$$\mu_1=\max(\boldsymbol{v}_1),\ \boldsymbol{u}_1=\boldsymbol{v}_1/\mu_1$$

(2) $k=2,3,\cdots$

1) 解 $\boldsymbol{L}\boldsymbol{y}_k=\boldsymbol{P}\boldsymbol{u}_{k-1}$ 求 $\boldsymbol{y}_k$

解 $\boldsymbol{U}\boldsymbol{v}_k=\boldsymbol{y}_k$ 求 $\boldsymbol{v}_k$

2) $\mu_k=\max(\boldsymbol{v}_k)$

3) 计算 $\boldsymbol{u}_k=\boldsymbol{v}_k/\mu_k$

例 6 用反幂法求

$$\boldsymbol{A}=\begin{pmatrix}2&1&0\\1&3&1\\0&1&4\end{pmatrix}$$

的对应于计算特征值 $\lambda=1.2679$(精确特征值为 $\lambda_3=3-\sqrt{3}$)的特征向量(用 5 位浮点数进行运算).

解 用部分选主元的三角分解将 $\boldsymbol{A}-p\boldsymbol{I}$(其中 $p=1.2679$)分

解为

$$P(A-pI)=LU,$$

其中

$$L=\begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0.7321 & -0.26807 & 1\end{pmatrix},$$

$$U=\begin{pmatrix}1 & 1.7321 & 1\\ 0 & 1 & 2.7321\\ 0 & 0 & 0.29405\times 10^{-3}\end{pmatrix},$$

$$P=\begin{pmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ 1 & 0 & 0\end{pmatrix}.$$

由 $Uv_1=(1,1,1)^{\mathrm T}$，得

$$v_1=(12692,-9290.3,3400.8)^{\mathrm T},$$

$$u_1=(1,-0.73198,0.26795)^{\mathrm T},$$

由 $LUv_2=Pu_1$，得

$$v_2=(20404,-14937,5467.4)^{\mathrm T},$$

$$u_2=(1,-0.73206,0.26796)^{\mathrm T},$$

λ_3 对应的特征向量是

$$x_3=(1,1-\sqrt{3},2-\sqrt{3})^{\mathrm T}\approx(1,-0.73205,0.26795)^{\mathrm T},$$

由此看出 u_2 是 x_3 的相当好的近似.

特征值 $\lambda_3\approx 1.2679+1/\mu_2=1.26794901$，$\lambda_3$ 的真值为 $\lambda_3=3-\sqrt{3}=1.26794912\cdots$.

8.3　豪斯霍尔德方法

8.3.1　引言

本节讨论两个问题

(1) 用初等反射阵作正交相似变换约化一般实矩阵 $\boldsymbol{A}$ 为上海森伯格阵.

(2) 用初等反射阵作正交相似变换约化对称矩阵 $\boldsymbol{A}$ 为对称三对角阵.

于是,求原矩阵特征值问题,就转化为求上海森伯格阵或对称三对角阵的特征值问题.

8.3.2 用正交相似变换约化一般矩阵为上海森柏格阵

设 $\boldsymbol{A}=(a_{ij})\in\mathbf{R}^{n\times n}$. 下面来说明,可选择初等反射阵 $\boldsymbol{U}_1$, $\boldsymbol{U}_2,\cdots,\boldsymbol{U}_{n-2}$ 使 $\boldsymbol{A}$ 经正交相似变换约化为一个上海森伯格阵.

(1) 设

$$\boldsymbol{A}=\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}=\begin{pmatrix} a_{11} & \boldsymbol{A}_{12}^{(1)} \\ \boldsymbol{c}_1 & \boldsymbol{A}_{22}^{(1)} \end{pmatrix},$$

其中 $\boldsymbol{c}_1=(a_{21},\cdots,a_{n1})^{\mathrm{T}}\in\mathbf{R}^{n-1}$,不妨设 $\boldsymbol{c}_1\neq\boldsymbol{0}$,否则这一步不需要约化. 于是,可选择初等反射阵 $\boldsymbol{R}_1=\boldsymbol{I}-\beta_1^{-1}\boldsymbol{u}_1\boldsymbol{u}_1^{\mathrm{T}}$ 使 $\boldsymbol{R}_1\boldsymbol{c}_1=-\sigma_1\boldsymbol{e}_1$,其中

$$\begin{cases} \sigma_1 = \operatorname{sgn}(a_{21})\left(\sum_{i=2}^{n} a_{i1}^2\right)^{1/2}, \\ \boldsymbol{u}_1 = \boldsymbol{c}_1+\sigma_1\boldsymbol{e}_1, \\ \beta_1 = \sigma_1(\sigma_1+a_{21}). \end{cases} \tag{3.1}$$

令

$$\boldsymbol{U}_1=\begin{pmatrix} 1 & \\ & \boldsymbol{R}_1 \end{pmatrix},$$

则

$$\boldsymbol{A}_2=\boldsymbol{U}_1\boldsymbol{A}_1\boldsymbol{U}_1=\begin{pmatrix} a_{11} & \boldsymbol{A}_{12}^{(1)}\boldsymbol{R}_1 \\ \boldsymbol{R}_1\boldsymbol{c}_1 & \boldsymbol{R}_1\boldsymbol{A}_{22}^{(1)}\boldsymbol{R}_1 \end{pmatrix}$$

$$
= \begin{pmatrix} a_{11} & a_{12}^{(2)} & a_{13}^{(2)} & \cdots & a_{1n}^{(2)} \\ -\sigma_1 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2n}^{(2)} \\ 0 & a_{32}^{(2)} & a_{33}^{(2)} & \cdots & a_{3n}^{(2)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & a_{n2}^{(2)} & a_{n3}^{(2)} & \cdots & a_{nn}^{(2)} \end{pmatrix} \equiv \begin{pmatrix} \boldsymbol{A}_{11}^{(2)} & \boldsymbol{A}_{12}^{(2)} \\ 0 \quad \boldsymbol{c}_2 & \boldsymbol{A}_{22}^{(2)} \end{pmatrix},
$$

其中 $\boldsymbol{c}_2=(a_{32}^{(2)},\cdots,a_{n2}^{(2)})^{\mathrm{T}}\in\mathbf{R}^{n-2}$，$\boldsymbol{A}_{22}^{(2)}\in\mathbf{R}^{(n-2)\times(n-2)}$.

(2) 第 k 步约化：重复上述过程，设对 $\boldsymbol{A}$ 已完成第 1 步，…，第 $k-1$ 步正交相似变换，即有

$$
\boldsymbol{A}_k=\boldsymbol{U}_{k-1}\boldsymbol{A}_{k-1}\boldsymbol{U}_{k-1},
$$

或

$$
\boldsymbol{A}_k=\boldsymbol{U}_{k-1}\cdots\boldsymbol{U}_1\boldsymbol{A}_1\boldsymbol{U}_1\cdots\boldsymbol{U}_{k-1},
$$

且

$$
\boldsymbol{A}_k=\begin{pmatrix} a_{11}^{(1)} & a_{12}^{(2)} & \cdots & a_{1,k-1}^{(k-1)} & a_{1k}^{(k)} & a_{1,k+1}^{(k)} & \cdots & a_{1n}^{(k)} \\ -\sigma_1 & a_{22}^{(2)} & \cdots & a_{2,k-1}^{(k-1)} & a_{2k}^{(k)} & a_{2,k+1}^{(k)} & & a_{2n}^{(k)} \\ & & \ddots & \vdots & \vdots & \vdots & & \vdots \\ & & & -\sigma_{k-1} & a_{kk}^{(k)} & a_{k,k+1}^{(k)} & \cdots & a_{kn}^{(k)} \\ & & & & a_{k+1,k}^{(k)} & a_{k+1,k+1}^{(k)} & \cdots & a_{k+1,n}^{(k)} \\ & & & & \vdots & \vdots & & \vdots \\ & & & & a_{nk}^{(k)} & a_{n,k+1}^{(k)} & \cdots & a_{nn}^{(k)} \end{pmatrix}
$$

$$
\begin{matrix} \quad k & \quad n-k \end{matrix}
$$

$$
\equiv \begin{pmatrix} \boldsymbol{A}_{11}^{(k)} & \boldsymbol{A}_{12}^{(k)} \\ \boldsymbol{0}\ \ \boldsymbol{c}_k & \boldsymbol{A}_{22}^{(k)} \end{pmatrix} \begin{matrix} k \\ n-k \end{matrix},
$$

其中 $\boldsymbol{c}_k=(a_{k+1,k}^{(k)},\cdots,a_{nk}^{(k)})^{\mathrm{T}}\in\mathbf{R}^{n-k}$，$\boldsymbol{A}_{11}^{(k)}$ 为 k 阶上海森伯格阵，$\boldsymbol{A}_{22}^{(k)}\in\mathbf{R}^{(n-k)\times(n-k)}$.

设 $\boldsymbol{c}_k\neq\boldsymbol{0}$，于是可选择初等反射阵 $\boldsymbol{R}_k$ 使 $\boldsymbol{R}_k\boldsymbol{c}_k=-\sigma_k\boldsymbol{e}_1$，其中，$\boldsymbol{R}_k$ 计算公式为

$$\begin{cases}\sigma_k = \operatorname{sgn}(a_{k+1,k}^{(k)})\left(\sum_{i=k+1}^{n}(a_{ik}^{(k)})^2\right)^{1/2}, \\ \boldsymbol{u}_k = \boldsymbol{c}_k + \sigma_k \boldsymbol{e}_1, \\ \beta_k = \sigma_k(a_{k+1,k}^{(k)} + \sigma_k), \\ \boldsymbol{R}_k = \boldsymbol{I} - \beta_k^{-1}\boldsymbol{u}_k\boldsymbol{u}_k^{\mathrm{T}}.\end{cases} \tag{3.2}$$

令
$$\boldsymbol{U}_k = \begin{pmatrix}\boldsymbol{I} & \\ & \boldsymbol{R}_k\end{pmatrix},$$

则
$$\boldsymbol{A}_{k+1} = \boldsymbol{U}_k\boldsymbol{A}_k\boldsymbol{U}_k = \left[\begin{array}{c:c} \boldsymbol{A}_{11}^{(k+1)} & \boldsymbol{A}_{12}^{(k)}\boldsymbol{R}_k \\ \hdashline \boldsymbol{0} \;\vdots\; \boldsymbol{R}_k\boldsymbol{c}_k & \boldsymbol{R}_k\boldsymbol{A}_{22}^{(k)}\boldsymbol{R}_k\end{array}\right] = \left[\begin{array}{c:c} \boldsymbol{A}_{11}^{(k+1)} & \boldsymbol{A}_{12}^{(k+1)} \\ \hdashline \boldsymbol{0} \;\vdots\; \boldsymbol{c}_{k+1} & \boldsymbol{A}_{22}^{(k+1)}\end{array}\right], \tag{3.3}$$

其中 $\boldsymbol{A}_{11}^{(k+1)}$ 为 $k+1$ 阶上海森伯格阵. 第 k 步约化只需计算 $\boldsymbol{A}_{12}^{(k)}\boldsymbol{R}_k$ 及 $\boldsymbol{R}_k\boldsymbol{A}_{22}^{(k)}\boldsymbol{R}_k$（当 $\boldsymbol{A}$ 为对称阵时,只需计算 $\boldsymbol{R}_k\boldsymbol{A}_{22}^{(k)}\boldsymbol{R}_k$).

(3) 重复上述过程,则有

$$\boldsymbol{U}_{n-2}\cdots\boldsymbol{U}_2\boldsymbol{U}_1\boldsymbol{A}\boldsymbol{U}_1\boldsymbol{U}_2\cdots\boldsymbol{U}_{n-2}$$

$$= \begin{bmatrix} a_{11} & * & * & \cdots & * & * \\ -\sigma_1 & a_{22}^{(2)} & * & \cdots & * & * \\ & -\sigma_2 & a_{33}^{(3)} & \cdots & * & * \\ & & \ddots & \ddots & \vdots & \vdots \\ & & & -\sigma_{n-2} & a_{n-1,n-1}^{(n-2)} & * \\ & & & & -\sigma_{n-1} & a_{nn}^{(n-1)}\end{bmatrix}$$

$$= \boldsymbol{A}_{n-1}.$$

总结上述讨论,有

定理 17（豪斯霍尔德约化矩阵为上海森伯格阵） 设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$,则存在初等反射阵 $\boldsymbol{U}_1,\boldsymbol{U}_2,\cdots,\boldsymbol{U}_{n-2}$ 使

$$\boldsymbol{U}_{n-2}\cdots\boldsymbol{U}_2\boldsymbol{U}_1\boldsymbol{A}\boldsymbol{U}_1\boldsymbol{U}_2\cdots\boldsymbol{U}_{n-2} \equiv \boldsymbol{U}_0^{\mathrm{T}}\boldsymbol{A}\boldsymbol{U}_0 = \boldsymbol{H}\text{（上海森伯格阵）}.$$

算法 1（豪斯霍尔德约化矩阵为上海森伯格型） 设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$,

本算法计算 $U_0^{\mathrm{T}}AU_0=H$（上海森伯格型），其中 $U_0=U_1U_2\cdots U_{n-2}$ 为初等反射阵的乘积.

1. $U_0 \leftarrow I$

2. 对于 $k=1,2,\cdots,n-2$

(1) 计算初等反射阵 R_k 使

$$R_k c_k = -\sigma_k e_1$$

(2) 约化计算

$$A \leftarrow U_k A U_k, \ U_k = \begin{pmatrix} I_k & \\ & R_k \end{pmatrix}$$

(3) $U_0 \leftarrow U_0 U_k$

本算法约需要 $\frac{5}{3}n^3$ 次乘法运算，要明显形成 U_0 还需要附加 $\frac{2}{3}n^3$ 次乘法.

例 7　用豪斯霍尔德方法将

$$A = A_1 = \begin{pmatrix} -4 & -3 & -7 \\ 2 & 3 & 2 \\ 4 & 2 & 7 \end{pmatrix}$$

矩阵约化为上 Hessenberg 阵.

解　选取初等反射阵 R_1 使 $R_1c_1=-\sigma_1e_1$，其中 $c_1=(2,4)^{\mathrm{T}}$.

(1) 计算 R_1：$\alpha=\max(2,4)=4$，$c_1 \to c_1'=(0.5, 1)^{\mathrm{T}}$（规范化）

$$\begin{cases} \sigma = \sqrt{1.25} = 1.118034, \\ u_1 = c_1' + \sigma e_1 = (1.618034, 1)^{\mathrm{T}}, \\ \beta_1 = \sigma(\sigma + 0.5) = 1.809017, \\ \sigma_1 = \alpha\sigma = 4.472136, \\ R_1 = I - \beta_1^{-1} u_1 u_1^{\mathrm{T}}. \end{cases}$$

则有

$$R_1 c_1 = -\sigma_1 e_1.$$

(2) 约化计算：

令
$$U_1=\begin{pmatrix}1 & \mathbf{0}\\ \mathbf{0} & R_1\end{pmatrix},$$
则
$$A_2=U_1AU_1=\begin{bmatrix}-4 & 7.602631 & -0.447214\\ -4.472136 & 7.799999 & -0.400000\\ 0 & -0.399999 & 2.200000\end{bmatrix}=H.$$

8.3.3 用正交相似变换约化对称阵为对称三对角阵

定理 18（豪斯霍尔德约化对称阵为对称三对角阵） 设 $A\in\mathbf{R}^{n\times n}$ 为对称矩阵，则存在初等反射阵 $U_1,U_2,\cdots,U_{n-2}$ 使

$$U_{n-2}\cdots U_2U_1AU_1U_2\cdots U_{n-2}=\begin{bmatrix}c_1 & b_1 & & & \\ b_1 & c_2 & b_2 & & \\ & \ddots & \ddots & \ddots & \\ & & b_{n-2} & c_{n-1} & b_{n-1}\\ & & & b_{n-1} & c_n\end{bmatrix}\equiv C.$$

证明 由定理 17，存在初等反射阵 $U_1,U_2,\cdots,U_{n-2}$ 使 $U_{n-2}\cdots U_2U_1AU_1U_2\cdots U_{n-2}=H=A_{n-1}$ 为上海森伯格阵，且 A_{n-1} 亦是对称阵，因此，A_{n-1} 为对称三对角阵.

由上面讨论可知，当 A 为对称阵时，由 $A_k\to A_{k+1}=U_kA_kU_k$ 一步约化计算中只需计算 R_k 及 $R_kA_{22}^{(k)}R_k$. 又由于 A 的对称性，故只需计算 $R_kA_{22}^{(k)}R_k$ 的对角线以下元素. 注意到

$$R_kA_{22}^{(k)}R_k=(I-\beta_k^{-1}u_ku_k^{\mathrm{T}})(A_{22}^{(k)}-\beta_k^{-1}A_{22}^{(k)}u_ku_k^{\mathrm{T}}).$$

引进记号

$$r_k=\beta_k^{-1}A_{22}^{(k)}u_k\quad\in\mathbf{R}^{n-k},$$
$$t_k=r_k-\frac{\beta_k^{-1}}{2}(u_k^{\mathrm{T}}r_k)u_k\quad\in\mathbf{R}^{n-k},$$

则
$$R_kA_{22}^{(k)}R_k=A_{22}^{(k)}-u_kt_k^{\mathrm{T}}-t_ku_k^{\mathrm{T}}$$
$$(i=k+1,\cdots,n,j=k+1,\cdots,i).$$

算法 2（豪斯霍尔德约化对称阵为对称三对角阵） 设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 为对称阵，本算法确定初等反射阵 $\boldsymbol{U}_k(k=1,\cdots,n-2)$ 使 $\boldsymbol{U}_{n-2}\cdots\boldsymbol{U}_1\boldsymbol{A}\boldsymbol{U}_1\cdots\boldsymbol{U}_{n-2}=\boldsymbol{C}$（为对称三对角阵）. $\boldsymbol{C}$ 的对角元 $c_i(i=1,\cdots,n)$ 存放在数组 $c(n)$ 内，$\boldsymbol{C}$ 的次对角元素 $b_i(i=1,\cdots,n-1)$ 存放在数组 $b(n)$ 内. 数组 $b(n)$ 最初可用来存放 $\boldsymbol{r}_k$ 及 $\boldsymbol{t}_k$，确定 $\boldsymbol{R}_k$ 中向量 $\boldsymbol{u}_k$ 的分量存放在 $\boldsymbol{A}$ 的相应位置. β_k 冲掉 a_{kk}，约化 $\boldsymbol{A}$ 的结果冲掉 $\boldsymbol{A}$，数组 $\boldsymbol{A}$ 的上部分元素不变. 如果第 k 步不需要变换则置 β_k 为零.

1. 对于 $k=1,2,\cdots,n-2$

(1) $c_k \leftarrow a_{kk}$

(2) 确定变换 $\boldsymbol{R}_k$

1) 计算 $d=\max\limits_{k+1\leqslant i\leqslant n}|a_{ik}|$

2) 如果 $d=0$，则 $a_{kk}\leftarrow\beta_k=0$，$b_k\leftarrow 0$，转 4)

3) 计算 $a_{ik}\leftarrow u_{ik}=a_{ik}/d \quad (i=k+1,\cdots,n)$

4) $\sigma=\operatorname{sgn}(u_{k+1,k})\sqrt{\sum\limits_{i=k+1}^{n}u_{ik}^2}$

5) $u_{k+1,k}\leftarrow u_{k+1,k}+\sigma$

6) $a_{kk}\leftarrow\beta_k=\sigma * u_{k+1,k}$

7) $b_k\leftarrow\sigma_k=-\sigma * d$

(3) 应用变换

1) $s\leftarrow 0$

2) 计算 $\boldsymbol{A}_{22}^{(k)}\boldsymbol{u}_k$ 及 $\boldsymbol{u}_k^{\mathrm{T}}\boldsymbol{r}_k$

对于 $i=k+1,\cdots,n$

(a) $b_i\leftarrow h=\sum\limits_{j=k+1}^{i}a_{ij} * u_{jk}+\sum\limits_{j=i+1}^{n}a_{ji} * u_{jk}$

(b) $s\leftarrow s+h * u_{ik}$

3) 计算 t_k

$$b_i\leftarrow(b_i-(s * u_{ik})/(2 * \beta_k))/\beta_k$$

$$(i = k+1, \cdots, n)$$

4）计算 $\boldsymbol{R}_k\boldsymbol{A}_{22}^{(k)}\boldsymbol{R}_k$ 对角线以下部分

$$\text{对于} \quad i = k+1, \cdots, n, j = k+1, \cdots, i$$

$$a_{ij} \leftarrow a_{ij} - u_{ik} * b_j - b_i * u_{jk}$$

5）继续循环 k

2. $c_{n-1} \leftarrow a_{n-1,n-1}$

$c_n \leftarrow a_{nn}$

$b_{n-1} \leftarrow a_{n,n-1}$

对对称阵 $\boldsymbol{A}$ 用初等反射阵正交相似约化为对称三对角阵大约需要$\frac{2}{3}n^3$ 次乘法.

用正交矩阵进行相似约化有一些特点，如构造的 $\boldsymbol{U}_k$ 容易求逆，且 $\boldsymbol{U}_k$ 的元素数量级不大，这个算法是十分稳定的.

8.4 QR 方法

8.4.1 QR 算法

Rutishauser(1958)利用矩阵的三角分解提出了计算矩阵特征值的 LR 算法，Francis(1961，1962)利用矩阵的 QR 分解建立了计算矩阵特征值的 QR 方法.

QR 方法是一种变换方法，是计算一般矩阵（中小型矩阵）全部特征值问题的最有效方法之一.

目前 QR 方法主要用来计算：(1)上海森伯格阵的全部特征值问题，(2)计算对称三对角矩阵的全部特征值问题，且 QR 方法具有收敛快，算法稳定等特点.

对于一般矩阵 $\boldsymbol{A} \in \mathbf{R}^{n\times n}$（或对称矩阵），则首先用豪斯霍尔德方法将 $\boldsymbol{A}$ 化为上海森伯格阵 $\boldsymbol{B}$（或对称三对角阵），然后再用 QR

方法计算 $\boldsymbol{B}$ 的全部特征值.

设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$，且对 $\boldsymbol{A}$ 进行 QR 分解，即

$$\boldsymbol{A}=\boldsymbol{QR},$$

其中 $\boldsymbol{R}$ 为上三角阵，$\boldsymbol{Q}$ 为正交阵，于是可得到一新矩阵

$$\boldsymbol{B}=\boldsymbol{RQ}=\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{AQ}.$$

显然，$\boldsymbol{B}$ 是由 $\boldsymbol{A}$ 经过正交相似变换得到，因此 $\boldsymbol{B}$ 与 $\boldsymbol{A}$ 特征值相同. 再对 $\boldsymbol{B}$ 进行 QR 分解，又可得一新的矩阵，重复这一过程可得到矩阵序列：

设 $\boldsymbol{A}=\boldsymbol{A}_1$

将 $\boldsymbol{A}_1$ 进行 QR 分解 $\boldsymbol{A}_1=\boldsymbol{Q}_1\boldsymbol{R}_1$

作矩阵 $\boldsymbol{A}_2=\boldsymbol{R}_1\boldsymbol{Q}_1=\boldsymbol{Q}_1^{\mathrm{T}}\boldsymbol{A}_1\boldsymbol{Q}_1$

…

求得 $\boldsymbol{A}_k$ 后将 $\boldsymbol{A}_k$ 进行 QR 分解 $\boldsymbol{A}_k=\boldsymbol{Q}_k\boldsymbol{R}_k$

形成矩阵 $\boldsymbol{A}_{k+1}=\boldsymbol{R}_k\boldsymbol{Q}_k=\boldsymbol{Q}_k^{\mathrm{T}}\boldsymbol{A}_k\boldsymbol{Q}_k$

…

QR 算法，就是利用矩阵的 QR 分解，按上述递推法则构造矩阵序列 $\{\boldsymbol{A}_k\}$ 的过程. 只要 $\boldsymbol{A}$ 为非奇异矩阵，则由 QR 算法就完全确定 $\{\boldsymbol{A}_k\}$.

定理 19（基本 QR 方法） 设 $\boldsymbol{A}=\boldsymbol{A}_1\in\mathbf{R}^{n\times n}$. 构造 QR 算法：

$$\begin{cases}\boldsymbol{A}_k=\boldsymbol{Q}_k\boldsymbol{R}_k & (\text{其中 } \boldsymbol{Q}_k^{\mathrm{T}}\boldsymbol{Q}_k=\boldsymbol{I},\boldsymbol{R}_k \text{ 为上三角阵});\\ \boldsymbol{A}_{k+1}=\boldsymbol{R}_k\boldsymbol{Q}_k & (k=1,2,\cdots),\end{cases}\tag{4.1}$$

记 $\widetilde{\boldsymbol{Q}}_k=\boldsymbol{Q}_1\boldsymbol{Q}_2\cdots\boldsymbol{Q}_k$，$\widetilde{\boldsymbol{R}}_k\equiv\boldsymbol{R}_k\cdots\boldsymbol{R}_2\boldsymbol{R}_1$，则有

(1) $\boldsymbol{A}_{k+1}$ 相似于 $\boldsymbol{A}_k$，即 $\boldsymbol{A}_{k+1}=\boldsymbol{Q}_k^{\mathrm{T}}\boldsymbol{A}_k\boldsymbol{Q}_k$；

(2) $\boldsymbol{A}_{k+1}=(\boldsymbol{Q}_1\boldsymbol{Q}_2\cdots\boldsymbol{Q}_k)^{\mathrm{T}}\boldsymbol{A}_1(\boldsymbol{Q}_1\boldsymbol{Q}_2\cdots\boldsymbol{Q}_k)$
$=\widetilde{\boldsymbol{Q}}_k^{\mathrm{T}}\boldsymbol{A}_1\widetilde{\boldsymbol{Q}}_k$；

(3) $\boldsymbol{A}^k$ 的 QR 分解式为 $\boldsymbol{A}^k=\widetilde{\boldsymbol{Q}}_k\widetilde{\boldsymbol{R}}_k$.

证明 (1)，(2)显然，现证(3). 用归纳法，显然，当 $k=1$ 时有

$\boldsymbol{A}_1=\widetilde{\boldsymbol{Q}}_1\widetilde{\boldsymbol{R}}_1=\boldsymbol{Q}_1\boldsymbol{R}_1$，设 $\boldsymbol{A}^{k-1}$ 有分解式

$$\boldsymbol{A}^{k-1}=\widetilde{\boldsymbol{Q}}_{k-1}\widetilde{\boldsymbol{R}}_{k-1},$$

于是

$$\begin{aligned}\widetilde{\boldsymbol{Q}}_k\widetilde{\boldsymbol{R}}_k &= \boldsymbol{Q}_1\boldsymbol{Q}_2\cdots(\boldsymbol{Q}_k\boldsymbol{R}_k)\cdots\boldsymbol{R}_1\\ &= \boldsymbol{Q}_1\boldsymbol{Q}_2\cdots\boldsymbol{Q}_{k-1}\boldsymbol{A}_k\boldsymbol{R}_{k-1}\cdots\boldsymbol{R}_1\\ &= \widetilde{\boldsymbol{Q}}_{k-1}\boldsymbol{A}_k\widetilde{\boldsymbol{R}}_{k-1}\\ &= \boldsymbol{A}\widetilde{\boldsymbol{Q}}_{k-1}\widetilde{\boldsymbol{R}}_{k-1}=\boldsymbol{A}^k(\text{因为 } \boldsymbol{A}_k=\widetilde{\boldsymbol{Q}}_{k-1}^{\mathrm{T}}\boldsymbol{A}\widetilde{\boldsymbol{Q}}_{k-1}).\end{aligned}$$

由第 5 章定理 30 或定理 31 知，将 $\boldsymbol{A}_k$ 进行 QR 分解，即将 $\boldsymbol{A}_k$ 用正交变换（左变换）化为上三角矩阵

$$\boldsymbol{Q}_k^{\mathrm{T}}\boldsymbol{A}_k=\boldsymbol{R}_k,$$

其中 $\boldsymbol{Q}_k^{\mathrm{T}}=\boldsymbol{P}_{n-1}\cdots\boldsymbol{P}_2\boldsymbol{P}_1$，故

$$\boldsymbol{A}_{k+1}=\boldsymbol{Q}_k^{\mathrm{T}}\boldsymbol{A}_k\boldsymbol{Q}_k=\boldsymbol{P}_{n-1}\cdots\boldsymbol{P}_2\boldsymbol{P}_1\boldsymbol{A}_k\boldsymbol{P}_1^{\mathrm{T}}\boldsymbol{P}_2^{\mathrm{T}}\cdots\boldsymbol{P}_{n-1}^{\mathrm{T}}.$$

这就是说 $\boldsymbol{A}_{k+1}$ 可由 $\boldsymbol{A}_k$ 按下述方法求得：

(1) 左变换 $\boldsymbol{P}_{n-1}\cdots\boldsymbol{P}_2\boldsymbol{P}_1\boldsymbol{A}_k=\boldsymbol{R}_k$（上三角阵）；

(2) 右变换 $\boldsymbol{R}_k\boldsymbol{P}_1^{\mathrm{T}}\boldsymbol{P}_2^{\mathrm{T}}\cdots\boldsymbol{P}_{n-1}^{\mathrm{T}}=\boldsymbol{A}_{k+1}$.

定理 20（QR 方法的收敛性） 设 $\boldsymbol{A}=(a_{ij})\in\mathbf{R}^{n\times n}$，

(1) 如果 $\boldsymbol{A}$ 的特征值满足：$|\lambda_1|>|\lambda_2|>\cdots>|\lambda_n|>0$；

(2) $\boldsymbol{A}$ 有标准型 $\boldsymbol{A}=\boldsymbol{X}\boldsymbol{D}\boldsymbol{X}^{-1}$ 其中 $\boldsymbol{D}=\mathrm{diag}(\lambda_1,\lambda_2,\cdots,\lambda_n)$，且设 $\boldsymbol{X}^{-1}$ 有三角分解 $\boldsymbol{X}^{-1}=\boldsymbol{L}\boldsymbol{U}$（$\boldsymbol{L}$ 为单位下三角阵，$\boldsymbol{U}$ 为上三角阵，则由 QR 算法产生的 $\{\boldsymbol{A}_k\}$ 本质上收敛于上三角矩阵，即

$$\boldsymbol{A}_k \xrightarrow{\text{本质上}} \boldsymbol{R}=\begin{pmatrix}\lambda_1 & * & \cdots & *\\ & \lambda_2 & \cdots & *\\ & & \ddots & \vdots\\ & & & \lambda_n\end{pmatrix}\quad(\text{当 } k\to\infty \text{ 时})$$

若记 $\boldsymbol{A}_k=(a_{ij}^{(k)})$，则

(1) $\lim\limits_{k\to\infty}a_{ii}^{(k)}=\lambda_i$； (4.2)

(2) 当 $i>j$ 时，$\lim\limits_{k\to\infty} a_{ij}^{(k)}=0$；　　(4.3)

当 $i<j$ 时 $a_{ij}^{(k)}$ 极限不一定存在.

证明可参阅[1].

定理 21　如果对称矩阵 $\boldsymbol{A}$ 满足定理 20 的条件，则由 QR 算法产生的 $\{\boldsymbol{A}_k\}$ 收敛于对角阵 $\boldsymbol{D}=\mathrm{diag}(\lambda_1,\lambda_2,\cdots,\lambda_n)$.

证明　由定理 20 即知.

关于 QR 算法收敛性的进一步结果为：

设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$，且 $\boldsymbol{A}$ 有完备的特征向量集合，如果 $\boldsymbol{A}$ 的等模特征值中只有实重特征值或多重复的共轭特征值，则由 QR 算法产生的 $\{\boldsymbol{A}_k\}$ 本质收敛于分块上三角矩阵(对角块为一阶和二阶子块)且对角块中每一个 2×2 子块给出 $\boldsymbol{A}$ 的一对共轭复特征值，每一个一阶对角子块给出 $\boldsymbol{A}$ 的实特征值，即

$$\boldsymbol{A}_k\to\begin{pmatrix}\lambda_1 & \cdots & * & * & * & \cdots & * & * \\ & \ddots & \vdots & \vdots & \vdots & & \vdots & \vdots \\ & & \lambda_m & * & * & \cdots & * & * \\ & & & \boldsymbol{B}_1 & & \cdots & * & * \\ & & & & & \ddots & \vdots & \\ & & & & & & \boldsymbol{B}_l & \end{pmatrix},$$

其中 $m+2l=n$，$\boldsymbol{B}_i$ 为 2×2 子块，它给出 $\boldsymbol{A}$ 一对共轭特征值.

8.4.2　带原点位移的 QR 方法

经分析指出：定理 20 中 $\lim\limits_{k\to\infty} a_{nn}^{(k)}=\lambda_n$ 的速度依赖于比值 $r_n=|\lambda_n/\lambda_{n-1}|$，当 r_n 很小时，收敛较快. 如果 s 为 λ_n 的一个估计，且对 $\boldsymbol{A}-s\boldsymbol{I}$ 运用 QR 算法，则 $(n,n-1)$ 元素将以收敛因子 $|(\lambda_n-s)/(\lambda_{n-1}-s)|$ 线性收敛于零，(n,n) 元素将比在基本算法中收敛更快.

为此，为了加速收敛，选择数列$\{s_k\}$，按下述方法构造矩阵序列$\{\boldsymbol{A}_k\}$，称为带原点位移的 QR 算法.

设 $\boldsymbol{A}=\boldsymbol{A}_1\in\mathbf{R}^{n\times n}$

对 $\boldsymbol{A}_1-s_1\boldsymbol{I}$ 进行 QR 分解 $\boldsymbol{A}_1-s_1\boldsymbol{I}=\boldsymbol{Q}_1\boldsymbol{R}_1$

形成矩阵 $\boldsymbol{A}_2=\boldsymbol{R}_1\boldsymbol{Q}_1+s_1\boldsymbol{I}=\boldsymbol{Q}_1^{\mathrm{T}}(\boldsymbol{A}-s_1\boldsymbol{I})\boldsymbol{Q}_1+s_1\boldsymbol{I}$

$=\boldsymbol{Q}_1^{\mathrm{T}}\boldsymbol{A}_1\boldsymbol{Q}_1$

求得 $\boldsymbol{A}_k$ 后，将 $\boldsymbol{A}_k-s_k\boldsymbol{I}$ 进行 QR 分解

$$\boldsymbol{A}_k-s_k\boldsymbol{I}=\boldsymbol{Q}_k\boldsymbol{R}_k,\quad k=3,4,\cdots \tag{4.4}$$

形成矩阵

$$\boldsymbol{A}_{k+1}=\boldsymbol{R}_k\boldsymbol{Q}_k+s_k\boldsymbol{I}=\boldsymbol{Q}_k^{\mathrm{T}}\boldsymbol{A}_k\boldsymbol{Q}_k \tag{4.5}$$

如果令 $\widetilde{\boldsymbol{Q}}_k=\boldsymbol{Q}_1\boldsymbol{Q}_2\cdots\boldsymbol{Q}_k$，$\widetilde{\boldsymbol{R}}_k=\boldsymbol{R}_k\cdots\boldsymbol{R}_2\boldsymbol{R}_1$，则有 $\boldsymbol{A}_{k+1}=\widetilde{\boldsymbol{Q}}_k^{\mathrm{T}}\boldsymbol{A}\widetilde{\boldsymbol{Q}}_k$，并且矩阵$(\boldsymbol{A}-s_1\boldsymbol{I})(\boldsymbol{A}-s_2\boldsymbol{I})\cdots(\boldsymbol{A}-s_n\boldsymbol{I})\equiv\varphi(\boldsymbol{A})$有 QR 分解式

$$\varphi(\boldsymbol{A})=\widetilde{\boldsymbol{Q}}_k\widetilde{\boldsymbol{R}}_k.$$

在带位移 QR 方法中，每步并不需要形成 $\boldsymbol{Q}$ 和 $\boldsymbol{R}$，可按下面的方法计算：

首先用正交变换（左变换）将 $\boldsymbol{A}_k-s_k\boldsymbol{I}$ 化为上三角阵，即

$$\boldsymbol{P}_{n-1}\cdots\boldsymbol{P}_2\boldsymbol{P}_1(\boldsymbol{A}_k-s_k\boldsymbol{I})=\boldsymbol{R}_k$$

（当 $\boldsymbol{A}$ 为上海森伯格阵或对称三对角阵时，$\boldsymbol{P}_i$ 可为平面旋转阵），则

$$\boldsymbol{A}_{k+1}=\boldsymbol{P}_{n-1}\cdots\boldsymbol{P}_2\boldsymbol{P}_1(\boldsymbol{A}_k-s_k\boldsymbol{I})\boldsymbol{P}_1^{\mathrm{T}}\boldsymbol{P}_2^{\mathrm{T}}\cdots\boldsymbol{P}_{n-1}^{\mathrm{T}}+s_k\boldsymbol{I}.$$

下面考虑用 QR 方法计算上海森伯格阵的特征值.

设 $\boldsymbol{B}$ 为上海森伯格阵，即

$$\boldsymbol{B}=\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1n}\\ b_{21} & b_{22} & \cdots & b_{2n}\\ & \ddots & \ddots & \vdots\\ & & b_{n,n-1} & b_{nn}\end{pmatrix}.$$

如果 $b_{i+1,i}\neq 0(i=1,2,\cdots,n-1)$，则称 $\boldsymbol{B}$ 为不可约上海森伯格阵.

设 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$，由定理 17 可选正交阵 $\boldsymbol{U}_0$ 使 $\boldsymbol{H}=\boldsymbol{U}_0^{\mathrm{T}}\boldsymbol{A}\boldsymbol{U}_0$ 为上海森伯格阵，对 $\boldsymbol{H}$ 应用 QR 算法.

QR 算法：$\boldsymbol{H}=\boldsymbol{H}_1$

对于 $k=1,2,\cdots$

$$\left.\begin{aligned}\boldsymbol{H}_k &= \boldsymbol{Q}_k\boldsymbol{R}_k \qquad (\text{QR 分解})\\ \boldsymbol{H}_{k+1} &= \boldsymbol{R}_k\boldsymbol{Q}_k\end{aligned}\right\} \tag{4.6}$$

不失一般性，可假设由(4.6)迭代产生的每一个上海森伯格阵 $\boldsymbol{H}_k$ 都是不可约的，否则，若在某步有

$$\boldsymbol{H}_{k+1}=\begin{matrix} & p & n-p & \\ & \left(\begin{matrix}\boldsymbol{H}_{11}\\ \boldsymbol{0}\end{matrix}\right. & \left.\begin{matrix}\boldsymbol{H}_{12}\\ \boldsymbol{H}_{22}\end{matrix}\right) & \begin{matrix}p\\ n-p\end{matrix}\end{matrix}.$$

于是，这个问题就分离为 $\boldsymbol{H}_{11}$ 与 $\boldsymbol{H}_{22}$ 两个较小的问题. 当 $p=n-1$ 或 $n-2$ 时，有

$$\boldsymbol{H}_{k+1}=\begin{matrix} & n-1 & 1 & \\ & \left(\begin{matrix}\boldsymbol{H}_{11}\\ \boldsymbol{0}\end{matrix}\right. & \left.\begin{matrix}\boldsymbol{H}_{12}\\ h_{nn}^{(k+1)}\end{matrix}\right) & \begin{matrix}n-1\\ 1\end{matrix}\end{matrix}$$

或

$$\boldsymbol{H}_{k+1}=\begin{matrix} & n-2 & 2 & \\ & \left(\begin{matrix}\boldsymbol{H}_{11}\\ \\ \boldsymbol{0}\end{matrix}\right. & \left.\begin{matrix}\boldsymbol{H}_{12}\\ * \quad *\\ * \quad *\end{matrix}\right) & \begin{matrix}n-2\\ \\ 2\end{matrix}\end{matrix},$$

即可求出 $\boldsymbol{H}$ 的特征值 $\lambda_n=h_{nn}^{(k+1)}$ 或 λ_{n-1},λ_n（由 $\boldsymbol{H}_{k+1}$ 右下角二阶阵的特征值求得），且求 $\boldsymbol{H}$ 的其余特征值时，转化为降阶求 $\boldsymbol{H}_{11}$ 的特征值.

实际上，每当 $\boldsymbol{H}_{k+1}$ 的次对角元适当小时，就可进行分离. 例如，如果

$$|h_{p+1,p}|\leqslant\varepsilon(|h_{pp}|+|h_{p+1,p+1}|),$$

就把 $h_{p+1,p}$ 视为零. 一般取 $\varepsilon=10^{-t}$，其中 t 是计算中有效数字的

位数.

8.4.3 用单步 QR 方法计算上海森伯格阵特征值

上海森伯格阵的单步 QR 方法：选取 s_k 并设

$$\boldsymbol{H}=\begin{pmatrix} h_{11} & h_{12} & \cdots & h_{1n} \\ h_{21} & h_{22} & \cdots & h_{2n} \\ & \ddots & \ddots & \vdots \\ & & h_{n,n-1} & h_{nn} \end{pmatrix}=\boldsymbol{H}_1 \quad (\text{设 } \boldsymbol{H} \text{ 为不可约阵}).$$

对于 $k=1,2,\cdots$ （用位移来加速收敛）

$$\begin{cases} \boldsymbol{H}_k - s_k\boldsymbol{I} = \boldsymbol{Q}_k\boldsymbol{R}_k \\ \boldsymbol{H}_{k+1} = \boldsymbol{R}_k\boldsymbol{Q}_k + s_k\boldsymbol{I} \end{cases}$$

由 $\boldsymbol{H}_k \to \boldsymbol{H}_{k+1}$ 实际计算为

(1) 左变换：$\boldsymbol{P}_{n-1,n}\cdots\boldsymbol{P}_{23}\boldsymbol{P}_{12}(\boldsymbol{H}_1-s_1\boldsymbol{I})=\boldsymbol{R}_1$（上三角阵）.

(2) 右变换：$\boldsymbol{H}_2=\boldsymbol{R}_1\boldsymbol{P}_{12}^{\mathrm{T}}\boldsymbol{P}_{23}^{\mathrm{T}}\cdots\boldsymbol{P}_{n-1,n}^{\mathrm{T}}+s_1\boldsymbol{I}$.

其中 $\boldsymbol{P}_{k,k+1}=\boldsymbol{P}(k,k+1)$ 为平面旋转阵.

(1) 左变换计算

$$h_{kk} \leftarrow h_{kk}-s_1 \quad (k=1,2,\cdots,n),$$

确定平面旋转阵 $\boldsymbol{P}_{12}=\boldsymbol{P}(1,2)$ 使

$$\boldsymbol{P}_{12}(\boldsymbol{H}_1-s_1\boldsymbol{I})=\begin{pmatrix} r_{11} & h_{12}^{(2)} & h_{13}^{(2)} & \cdots & h_{1n}^{(2)} \\ 0 & h_{22}^{(2)} & h_{23}^{(2)} & \cdots & h_{2n}^{(2)} \\ 0 & h_{32} & h_{33} & \cdots & h_{3n} \\ & & \ddots & \ddots & \vdots \\ & & & h_{n,n-1} & h_{nn} \end{pmatrix}.$$

设已完成第 1 次，…第 $k-1$ 次左变换，即有

$$\boldsymbol{P}_{k-1,k}\cdots\boldsymbol{P}_{23}\boldsymbol{P}_{12}(\boldsymbol{H}_1-s_1\boldsymbol{I})=$$

$$\begin{bmatrix} r_{11} & \cdots & h_{1,k-1}^{(2)} & h_{1k}^{(2)} & \cdots & h_{1n}^{(2)} \\ & \ddots & \vdots & \vdots & & \vdots \\ & & r_{k-1,k-1} & h_{k-1,k}^{(k)} & \cdots & h_{k-1,n}^{(k)} \\ & & & h_{kk}^{(k)} & \cdots & h_{kn}^{(k)} \\ & & & h_{k+1,k} & \cdots & h_{k+1,n} \\ & & & & \ddots & \vdots \\ & & & & h_{n,n-1} & h_{nn} \end{bmatrix}. \tag{4.7}$$

确定平面旋转阵 $\boldsymbol{P}_{k,k+1}=\boldsymbol{P}(k,k+1)$，使 $h_{k+1,k}$ 变为 0，且完成第 k 次左变换 $\boldsymbol{P}_{k,k+1}(\boldsymbol{P}_{k-1,k}\cdots\boldsymbol{P}_{12}(\boldsymbol{H}_1-s_1\boldsymbol{I}))$ 计算（只需计算(4.7)阵第 k 行及第 $k+1$ 行元素）.

继续这一过程，最后有

$$\boldsymbol{P}_{n-1,n}\cdots\boldsymbol{P}_{12}(\boldsymbol{H}_1-s_1\boldsymbol{I})=\boldsymbol{R}_1 \quad（上三角阵）.$$

(2) 右变换计算

$$\boldsymbol{H}_2=\boldsymbol{R}_1\boldsymbol{P}_{12}^{\mathrm{T}}\boldsymbol{P}_{23}^{\mathrm{T}}\cdots\boldsymbol{P}_{n-1,n}^{\mathrm{T}}+s_1\boldsymbol{I},$$

在第 k 次右变换 $(\boldsymbol{R}_1\boldsymbol{P}_{12}^{\mathrm{T}}\cdots)\boldsymbol{P}_{k,k+1}^{\mathrm{T}}$ 中，只需计算 $\boldsymbol{R}_1\boldsymbol{P}_{12}^{\mathrm{T}}\cdots\boldsymbol{P}_{k-1,k}^{\mathrm{T}}$ 第 k 列及第 $k+1$ 列元素.

$$h_{k,k}\leftarrow h_{k,k}+s_1 \qquad (k=1,2,\cdots,n).$$

最后

$$\boldsymbol{H}_2=\boldsymbol{R}_1\boldsymbol{P}_{12}^{\mathrm{T}}\cdots\boldsymbol{P}_{n-1,n}^{\mathrm{T}}+s_1\boldsymbol{I}$$

$$=\begin{bmatrix} * & * & \cdots & * \\ * & * & \cdots & * \\ & \ddots & \ddots & \vdots \\ & & * & * \end{bmatrix}（为上海森伯格阵）.$$

由上述讨论指出，如果 $\boldsymbol{H}\in\mathbf{R}^{n\times n}$ 为上海森伯格阵，则用 QR 算法产生的 $\boldsymbol{H}_2,\boldsymbol{H}_3,\cdots,\boldsymbol{H}_k,\cdots$ 亦是上海森伯格阵. 即上海森伯格阵在 QR 变换下形式不变.

下述定理讨论一个极端的情况

定理 22　设：(1) $\boldsymbol{H}\in\mathbf{R}^{n\times n}$ 为不可约上海森伯格阵；(2) μ 为

$H=H_1$ 一个特征值. 则 QR 方法

$$\begin{cases} H_1 - \mu I = QR & \text{(QR 分解)} \\ H_2 = RQ + \mu I \end{cases}$$

中 $h_{n,n-1}^{(2)}=0$, $h_{nn}^{(2)}=\mu$.

证明 记

$$R = \begin{pmatrix} r_{11} & \cdots & r_{1n} \\ & \ddots & \vdots \\ & & r_{nn} \end{pmatrix} \quad \text{(上三角阵)}.$$

由设 H_1 为不可约阵,则上海森伯格阵 $H_1-\mu I$ 亦为不可约. 由将上海森伯格阵 $H_1-\mu I$ 约化为上三角阵 R 的平面旋转变换的取法可知

$$|r_{ii}| \geqslant |h_{i+1,i}| \neq 0 \quad (i=1,2,\cdots,n-1),$$

又因为 $Q^{\mathrm{T}}(H_1-\mu I)=R$ 为奇异矩阵,从而得到 $r_{nn}=0$. 因此,H_2 的最后一行为$(0,0,\cdots,0,\mu)$,即

$$h_{nn-1}^{(2)}=0,\ h_{nn}^{(2)}=\mu.$$

这就启发我们在 QR 方法迭代中,参数 s_k 可选为 $h_{nn}^{(k)}$,即 H_k 的(n,n)元素. 通常可以作为特征值的最好近似.

算法 3(上海森伯格阵的 QR 算法) 给定 $H\in \mathbf{R}^{n\times n}$ 为上海森伯格阵,本算法计算

$$\begin{cases} H_1 - sI = Q_1R_1 & \text{(QR 分解)} \quad (\text{取 } s = h_{nn}) \\ H_2 = R_1Q_1 + sI \end{cases}$$

且 H_2 覆盖 $H(H=H_1)$

1. $h_{11}\leftarrow h_{11}-s$

2. 对于 $k=1,2,\cdots,n-1$

(1) $h_{k+1,k+1}\leftarrow h_{k+1,k+1}-s$

(2) 确定 $P(k,k+1)$使

$$\begin{pmatrix} c_k & s_k \\ -s_k & c_k \end{pmatrix}\begin{pmatrix} h_{kk} \\ h_{k+1,k} \end{pmatrix} = \begin{pmatrix} r_{kk} \\ 0 \end{pmatrix}$$

(3) 左变换

对于 $j=k,\cdots,n$

$$\begin{pmatrix} h_{kj} \\ h_{k+1,j} \end{pmatrix} \leftarrow \begin{pmatrix} c_k & s_k \\ -s_k & c_k \end{pmatrix} \begin{pmatrix} h_{kj} \\ h_{k+1,j} \end{pmatrix}$$

3. 对于 $k=1,2,\cdots,n-1$

(1) 右变换

对于 $i=1,2,\cdots,k+1$

$$(h_{ik}, h_{i,k+1}) \leftarrow (h_{ik}, h_{i,k+1}) \begin{pmatrix} c_k & -s_k \\ s_k & c_k \end{pmatrix}$$

(2) $h_{kk} \leftarrow h_{kk} + s$

4. $h_{nn} \leftarrow h_{nn} + s$

如果用不同的位移 $s_k = h_{nn}^{(k)}$，反复应用算法 3 就产生正交相似的上海森伯格阵序列 $\boldsymbol{H}_1, \boldsymbol{H}_2, \cdots, \boldsymbol{H}_k, \cdots$. 当 $h_{n,n-1}^{(k)}$ 充分小时，可将它置为零就得到 $\boldsymbol{H}$ 的近似特征值 $\lambda_n \approx h_{nn}^{(k)}$. 再将矩阵降阶，对较小矩阵连续应用算法.

例 8　用 QR 方法计算对称三对角矩阵

$$\boldsymbol{A} = \boldsymbol{A}_1 = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 3 & 1 \\ 0 & 1 & 4 \end{pmatrix}.$$

的全部特征值.

解　选取 $s_k = a_{nn}^{(k)}$，则 $s_1 = 4$.

$$\boldsymbol{P}_{23}\boldsymbol{P}_{12}(\boldsymbol{A}_1 - s_1\boldsymbol{I}) = \boldsymbol{R}$$

$$= \begin{pmatrix} 2.2361 & -1.342 & 0.4472 \\ & 1.0954 & -0.3651 \\ & & 0.81650 \end{pmatrix},$$

$$\boldsymbol{A}_2 = \boldsymbol{R}\boldsymbol{P}_{12}^{\mathrm{T}}\boldsymbol{P}_{23}^{\mathrm{T}} + s_1\boldsymbol{I}$$

$$= \begin{pmatrix} 1.4000 & 0.4899 & 0 \\ 0.4899 & 3.2667 & 0.7454 \\ 0 & 0.7454 & 4.3333 \end{pmatrix}.$$

$$\boldsymbol{A}_3 = \begin{pmatrix} 1.2915 & 0.2017 & 0 \\ 0.2017 & 3.0202 & 0.2724 \\ 0 & 0.2724 & 4.6884 \end{pmatrix},$$

$$\boldsymbol{A}_4 = \begin{pmatrix} 1.2737 & 0.0993 & 0 \\ 0.0993 & 2.9943 & 0.0072 \\ 0 & 0.0072 & 4.7320 \end{pmatrix},$$

$$\boldsymbol{A}_5 = \begin{pmatrix} 1.2694 & 0.0498 & 0 \\ 0.0498 & 2.9986 & 0 \\ 0 & 0 & 4.7321 \end{pmatrix},$$

$$\widetilde{\boldsymbol{A}}_5 = \begin{pmatrix} 1.2694 & 0.0498 \\ 0.0498 & 2.9986 \end{pmatrix}.$$

现在收缩,继续对 $\boldsymbol{A}_5$ 的子矩阵 $\widetilde{\boldsymbol{A}}_5 \in \mathbf{R}^{2\times 2}$ 进行变换,得到

$$\widetilde{\boldsymbol{A}}_6 = \boldsymbol{P}_{12}(\widetilde{\boldsymbol{A}}_5 - s_5\boldsymbol{I})\boldsymbol{P}_{12}^{\mathrm{T}} + s_5\boldsymbol{I} = \begin{pmatrix} 1.2680 & -4\times 10^{-5} \\ -4\times 10^{-5} & 3.0000 \end{pmatrix},$$

故求得 $\boldsymbol{A}$ 近似特征值为

$$\lambda_3 \approx 4.7321,\ \lambda_2 \approx 3.0000, \lambda_1 \approx 1.2680.$$

而 $\boldsymbol{A}$ 的特征值是

$$\lambda_3 = 3+\sqrt{3} \approx 4.7321, \lambda_2 = 3.0, \lambda_1 = 3-\sqrt{3} \approx 1.2679.$$

算法3是在实数中进行选择位移 $s_k = h_{nn}^{(k)}$,不能逼近一个复特征值,所以算法3不能用来计算 $\boldsymbol{H}$ 的复特征值.

8.4.4* 双步QR方法(隐式QR方法)

在本章第3节中将 $\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 经过正交相似变换化为上海森伯格矩阵 $\boldsymbol{H}$,即 $\boldsymbol{U}_0^{\mathrm{T}}\boldsymbol{A}\boldsymbol{U}_0 = \boldsymbol{H}$,其中 $\boldsymbol{H}$ 不是唯一的. 但是,如果规定了正交矩阵 $\boldsymbol{U}_0$ 的第一列,则 $\boldsymbol{U}_0$ 和 $\boldsymbol{H}$ 除差 ± 1 因子外唯一.

定理 23(隐式 Q 定理) 设 $A\in\mathbf{R}^{n\times n}$,且:

(1) $Q=(q_1 q_2\cdots q_n)$ 及 $V=(v_1 v_2\cdots v_n)$ 都是正交阵,且有 $Q^{\mathrm{T}}AQ=H$,$V^{\mathrm{T}}AV=G$ 都是上海森伯格阵.

(2) H 为不可约上海森伯格阵,且 $q_1=v_1$(即 Q 与 V 第 1 列相同). 则:

(1) $v_i=\pm q_i$, 且 $|h_{i,i-1}|=|g_{i,i-1}|$ $(i=2,\cdots,n)$;

(2) $G=D^{-1}HD$,其中 $D=\mathrm{diag}(1,\pm1,\cdots,\pm1)$,即 H 和 G 在 $G=D^{-1}HD$ 意义上"本质上相等".

算法 3 不能用来求 H 的一个复特征值,当 H(上海森伯格阵)的依模最小特征值是复数时,位移参数 s_k, s_{k+1} 可取为某步 H_k 右下角的二阶矩阵

$$G=\begin{pmatrix} h_{n-1,n-1} & h_{n-1,n} \\ h_{n,n-1} & h_{nn} \end{pmatrix} \tag{4.8}$$

的特征值.

当 G 的特征值 s_1 与 s_2 为复数时,如果应用算法 3 就要引进复数运算,这对于实矩阵 H 是不必要的,事实上,在某些条件下,可以用正交相似变换将 H 约化为实 Schur 型.

下面引进隐式位移的 QR 方法,即用 s_1 与 s_2 作位移连续进行二次单步的 QR 迭代,使用复位移,又避免复数运算.

(1) 设 $H=H_1\in\mathbf{R}^{n\times n}$ 为上海森伯格阵,取共轭复数 s_1,s_2 作两步位移的 QR 方法,即

$$\left.\begin{aligned} &H_1-s_1I=Q_1R_1,\\ &H_2=R_1Q_1+s_1I=Q_1^{\mathrm{T}}H_1Q_1,\\ &H_2-s_2I=Q_2R_2,\\ &H_3=R_2Q_2+s_2I=Q_2^{\mathrm{T}}Q_1^{\mathrm{T}}H_1Q_1Q_2=Q^{\mathrm{T}}H_1Q,\\ &\text{其中},Q=Q_1Q_2,\ R=R_2R_1. \end{aligned}\right\} \tag{4.9}$$

显然 $M=(H_1-s_1I)(H_1-s_2I)$ 有 QR 分解

$$M=QR. \tag{4.10}$$

事实上,由(4.9)式并利用 $\boldsymbol{H}_2-s_2\boldsymbol{I}=\boldsymbol{Q}_1^{\mathrm{T}}(\boldsymbol{H}_1-s_2\boldsymbol{I})\boldsymbol{Q}_1=\boldsymbol{Q}_2\boldsymbol{R}_2$ 有

$$\begin{aligned}\boldsymbol{M}&=(\boldsymbol{H}_1-s_2\boldsymbol{I})\boldsymbol{Q}_1\boldsymbol{R}_1\\&=(\boldsymbol{Q}_1\boldsymbol{Q}_2\boldsymbol{R}_2\boldsymbol{Q}_1^{\mathrm{T}})\boldsymbol{Q}_1\boldsymbol{R}_1\\&=\boldsymbol{Q}_1\boldsymbol{Q}_2\boldsymbol{R}_2\boldsymbol{R}_1=\boldsymbol{QR}.\end{aligned}$$

且 $\boldsymbol{M}$ 阵为实矩阵,这是因为(即使 $\boldsymbol{G}$ 特征值为复数)

$$\boldsymbol{M}=\boldsymbol{H}_1^2-(s_1+s_2)\boldsymbol{H}_1+s_1s_2\,\boldsymbol{I}, \tag{4.11}$$

其中 $s_1+s_2=h_{n-1,n-1}+h_{nn}=s$, $s_1s_2=h_{n-1,n-1}h_{nn}-h_{n,n-1}h_{n-1,n}=t$ 为实数. 于是,(4.10)式为实矩阵 $\boldsymbol{M}$ 的 QR 分解,并且可以选取 $\boldsymbol{Q}_1$ 和 $\boldsymbol{Q}_2$ 使 $\boldsymbol{Q}=\boldsymbol{Q}_1\boldsymbol{Q}_2$ 为实的正交阵. 由此得出

$$\boldsymbol{H}_3=(\boldsymbol{Q}_1\boldsymbol{Q}_2)^{\mathrm{T}}\boldsymbol{H}_1(\boldsymbol{Q}_1\boldsymbol{Q}_2)=\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{H}_1\boldsymbol{Q}$$

是实矩阵.

如果用下述算法就能保证 $\boldsymbol{H}_3$ 是实矩阵

(a) 直接形成实矩阵 $\boldsymbol{M}=\boldsymbol{H}_1^2-s\boldsymbol{H}_1+t\boldsymbol{I}$

(b) 计算 $\boldsymbol{M}$ 阵的实 QR 分解 $\boldsymbol{M}=\boldsymbol{QR}$

(c) 令 $\boldsymbol{H}_3=\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{H}_1\boldsymbol{Q}$

但是(a)需要 $O(n^3)$次乘法运算,不实用.

(2) 根据隐式 Q 定理,如果按下述算法进行,就有可能用 $O(n^2)$次运算来实现从 $\boldsymbol{H}_1$ 到 $\boldsymbol{H}_3$ 的转换.

(a′) 求与 $\boldsymbol{Q}$ 有相同第一列的正交阵 $\boldsymbol{P}_0$

(b′) 应用豪斯霍尔德方法将 $\boldsymbol{P}_0^{\mathrm{T}}\boldsymbol{H}_1\boldsymbol{P}_0$ 化为一个上海森伯格阵,即

$$\boldsymbol{P}_{n-2}\cdots\boldsymbol{P}_2\boldsymbol{P}_1(\boldsymbol{P}_0^{\mathrm{T}}\boldsymbol{H}_1\boldsymbol{P}_0)\boldsymbol{P}_1\boldsymbol{P}_2\cdots\boldsymbol{P}_{n-2}=\boldsymbol{H}'.$$

记 $\boldsymbol{Q}'=\boldsymbol{P}_0\boldsymbol{P}_1\cdots\boldsymbol{P}_{n-2}$, 上式为

$$\boldsymbol{Q}'^{\mathrm{T}}\boldsymbol{H}_1\boldsymbol{Q}'=\boldsymbol{H}'.$$

显然,$\boldsymbol{Q}'$的第一列与 $\boldsymbol{P}_0$ 的第一列相同,即 $\boldsymbol{Q}'$ 与 $\boldsymbol{Q}$ 第一列相同($\boldsymbol{Q}'\boldsymbol{e}_1=\boldsymbol{P}_0\boldsymbol{e}_1=\boldsymbol{Q}\boldsymbol{e}_1$). 若 $\boldsymbol{Q}^{\mathrm{T}}\boldsymbol{H}_1\boldsymbol{Q}$ 与 $\boldsymbol{Q}'^{\mathrm{T}}\boldsymbol{H}_1\boldsymbol{Q}'$两者都是不可约上海森伯格阵,则由隐式 Q 定理 $\boldsymbol{H}'$ 与 $\boldsymbol{H}_3$ 本质上相等.

(3) 如何寻求正交阵 $\boldsymbol{P}_0$.

由于 $\boldsymbol{M}=\boldsymbol{QR}$（为 $\boldsymbol{M}$ 的 QR 分解），则

$$\boldsymbol{M}\boldsymbol{e}_1 = \boldsymbol{Q}\boldsymbol{R}\boldsymbol{e}_1 = r_{11}\boldsymbol{Q}\boldsymbol{e}_1.$$

这说明 $\boldsymbol{Q}$ 的第一列即是 $\boldsymbol{M}$ 第一列的一个倍数，于是，对 $\boldsymbol{M}$ 阵的第一列（非零）寻求初等反射阵 $\boldsymbol{P}_0$ 使

$$\boldsymbol{P}_0(\boldsymbol{M}\boldsymbol{e}_1) = r_{11}\boldsymbol{e}_1 \quad (\text{其中 } r_{11} = -\sigma)$$

即

$$\boldsymbol{M}\boldsymbol{e}_1 = r_{11}\boldsymbol{P}_0\boldsymbol{e}_1.$$

这说明 $\boldsymbol{P}_0$ 与 $\boldsymbol{Q}$ 具有相同的第一列.

由于　$\boldsymbol{M}=(\boldsymbol{H}-s_1\boldsymbol{I})(\boldsymbol{H}-s_2\boldsymbol{I})$，则

$$\boldsymbol{M}\boldsymbol{e}_1 = (x,y,z,0,\cdots,0)^{\mathrm{T}}$$

其中

$$\left.\begin{aligned} x &= (h_{11}-s_1)(h_{11}-s_2)+h_{12}h_{21} \\ &= h_{11}^2+h_{12}h_{21}-s\,h_{11}+t, \\ y &= (h_{11}-s_2)h_{21}+(h_{22}-s_1)h_{21} \\ &= h_{21}(h_{11}+h_{22}-s), \\ z &= h_{21}h_{32}. \end{aligned}\right\} \tag{4.12}$$

双步 QR 方法：设 $\boldsymbol{H}=\boldsymbol{H}_1\in\mathbf{R}^{n\times n}$ 为不可约上海森伯格阵.

(a) 计算 $\boldsymbol{M}$ 阵的第一列. 即按(4.12)式计算

$$\boldsymbol{M}\boldsymbol{e}_1 = (x,y,z,0,\cdots,0)^{\mathrm{T}};$$

(b) 确定初等反射阵 $\boldsymbol{P}_0$ 使

$$\boldsymbol{P}_0(\boldsymbol{M}\boldsymbol{e}_1) = -\sigma\boldsymbol{e}_1,$$

即确定初等反射阵 $\boldsymbol{R}_0\in\mathbf{R}^{3\times3}$ 使 $\boldsymbol{R}_0\begin{pmatrix}x\\y\\z\end{pmatrix}=-\sigma\boldsymbol{e}_1$；

$$\boldsymbol{P}_0 = \begin{pmatrix}\boldsymbol{R}_0 & \\ & \boldsymbol{I}\end{pmatrix}\begin{matrix}3\\n-3\end{matrix};$$

(c) 计算初等反射阵 $\boldsymbol{P}_1,\boldsymbol{P}_2,\cdots,\boldsymbol{P}_{n-2}$ 使

$$\boldsymbol{P}_{n-2}\cdots\boldsymbol{P}_2\boldsymbol{P}_1(\boldsymbol{P}_0\boldsymbol{H}_1\boldsymbol{P}_0)\boldsymbol{P}_1\boldsymbol{P}_2\cdots\boldsymbol{P}_{n-2} = \boldsymbol{H}'$$

为上海森伯格阵，则 $\boldsymbol{Q}=\boldsymbol{Q}_1\boldsymbol{Q}_2$ 与 $\boldsymbol{Q}'=\boldsymbol{P}_0\boldsymbol{P}_1\cdots\boldsymbol{P}_{n-2}$ 第一列相同且 $\boldsymbol{H}'=\boldsymbol{H}_3$.

这样上面的算法就完成了从 $\boldsymbol{H}_1$ 到 $\boldsymbol{H}_3$ 的变换，但没有明显的应用到位移 s_1 和 s_2.

算法的具体实现可参看文献[1]及[19]，本算法在数学库中均有相应软件，可直接使用.

评　注

本章介绍了计算一般矩阵主特征值及对应特征向量的迭代法(幂法)，这种方法在计算过程中原始矩阵 $\boldsymbol{A}$ 始终不变. 因此，这种方法适于求高阶稀疏矩阵的特征值问题. 主特征值为复特征值的情况可参考文献[3].

反幂法是计算矩阵特征向量的一个有效方法，主要用来计算海森伯格阵或三对角矩阵的对应于一个给定的近似特征值的特征向量.

本章还介绍了用正交相似变换的方法约化一般矩阵 $\boldsymbol{A}$ 为上海森伯格阵的豪斯霍尔德方法，计算一般矩阵 $\boldsymbol{A}$ 全部特征值问题的 QR 方法. 这些方法都是利用矩阵的正交相似变换约化矩阵 $\boldsymbol{A}$ 为某种简单形式，以期求 $\boldsymbol{A}$ 的特征值的方法. 这种方法将破坏原始矩阵. QR 方法(与豪斯霍尔德方法结合使用)具有收敛快，精度高的特点，是求任意实矩阵(中，小型)特征值问题的最有效的方法之一. 对计算矩阵特征值问题有兴趣的读者可进一步参考文献[13],[19],[20].

习　题

1. 用幂法计算下列矩阵的主特征值及对应的特征向量：

$$(\mathrm{a})\ \boldsymbol{A}_1=\begin{pmatrix}7&3&-2\\3&4&-1\\-2&-1&3\end{pmatrix},\ (\mathrm{b})\ \boldsymbol{A}_2=\begin{pmatrix}3&-4&3\\-4&6&3\\3&3&1\end{pmatrix},$$

当特征值有 3 位小数稳定时迭代终止.

2. 利用反幂法求矩阵

$$\begin{pmatrix} 6 & 2 & 1 \\ 2 & 3 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

的最接近于 6 的特征值及对应的特征向量.

3. 求矩阵

$$\begin{pmatrix} 4 & 0 & 0 \\ 0 & 3 & 1 \\ 0 & 1 & 3 \end{pmatrix}$$

与特征值 4 对应的特征向量.

4. (a) 设 $\boldsymbol{A}$ 是对称矩阵，λ 和 $\boldsymbol{x}(\|\boldsymbol{x}\|_2=1)$是 $\boldsymbol{A}$ 的一个特征值及相应的特征向量. 又设 $\boldsymbol{P}$ 为一个正交阵，使

$$\boldsymbol{Px} = \boldsymbol{e}_1 = (1,0,\cdots,0)^{\mathrm{T}}.$$

证明 $\boldsymbol{B}=\boldsymbol{PAP}^{\mathrm{T}}$ 的第一行和第一列除了 λ 外其余元素均为零.

(b) 对于矩阵

$$\boldsymbol{A} = \begin{pmatrix} 2 & 10 & 2 \\ 10 & 5 & -8 \\ 2 & -8 & 11 \end{pmatrix},$$

$\lambda=9$ 是其特征值，$\boldsymbol{x}=\left(\frac{2}{3},\frac{1}{3},\frac{2}{3}\right)^{\mathrm{T}}$ 是相应于 9 的特征向量，试求一初等反射阵 $\boldsymbol{P}$，使 $\boldsymbol{Px}=\boldsymbol{e}_1$，并计算 $\boldsymbol{B}=\boldsymbol{PAP}^{\mathrm{T}}$.

5. 利用初等反射阵将

$$\boldsymbol{A} = \begin{pmatrix} 1 & 3 & 4 \\ 3 & 1 & 2 \\ 4 & 2 & 1 \end{pmatrix}$$

正交相似约化为对称三对角阵.

6. 设 $\boldsymbol{A}_{n-1}$是由豪斯霍尔德方法得到的矩阵，又设 $\boldsymbol{y}$ 是 $\boldsymbol{A}_{n-1}$ 的一个特征向量.

(a) 证明矩阵 $\boldsymbol{A}$ 对应的特征向量是 $\boldsymbol{x}=\boldsymbol{P}_1\boldsymbol{P}_2\cdots\boldsymbol{P}_{n-2}\boldsymbol{y}$，

(b) 对于给出的 $\boldsymbol{y}$ 应如何计算 $\boldsymbol{x}$？

7. 用带位移的 QR 方法计算

$$(a)\ \boldsymbol{A}=\begin{pmatrix}1 & 2 & 0\\ 2 & -1 & 1\\ 0 & 1 & 3\end{pmatrix},\quad (b)\ \boldsymbol{B}=\begin{pmatrix}3 & 1 & 0\\ 1 & 2 & 1\\ 0 & 1 & 1\end{pmatrix}$$

的全部特征值.

8. 试用初等反射阵将

$$\boldsymbol{A}=\begin{pmatrix}1 & 1 & 1\\ 2 & -1 & -1\\ 2 & -4 & 5\end{pmatrix}$$

分解为 $\boldsymbol{QR}$ 的形式,其中 $\boldsymbol{Q}$ 为正交阵,$\boldsymbol{R}$ 为上三角阵.

9. 设

$$\boldsymbol{A}=\begin{pmatrix}4 & -1 & & & \\ -1 & 4 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 4 & -1\\ & & & -1 & 4\end{pmatrix}\in \mathbf{R}^{n\times n},$$

试确定 $\boldsymbol{A}$ 及 $\boldsymbol{A}^{-1}$ 特征值的界.

10. 设 $\boldsymbol{A}=\begin{array}{c}\begin{matrix}3 & 2\end{matrix}\\ \begin{pmatrix}\boldsymbol{A}_{11} & \boldsymbol{A}_{12}\\ \boldsymbol{0} & \boldsymbol{A}_{22}\end{pmatrix}\begin{matrix}3\\2\end{matrix}\end{array}$,又设 λ_i 为 $\boldsymbol{A}_{11}$ 的特征值,λ_j 为 $\boldsymbol{A}_{22}$ 的特征值,$\boldsymbol{x}_i=(\alpha_1,\alpha_2,\alpha_3)^{\mathrm{T}}$ 为对应于 λ_i,$\boldsymbol{A}_{11}$ 的特征向量,$\boldsymbol{y}_i=(\beta_1,\beta_2)^{\mathrm{T}}$ 为对应于 λ_j,$\boldsymbol{A}_{22}$ 的特征向量.求证:

(1) λ_i,λ_j 为 $\boldsymbol{A}$ 的特征值.

(2) $\boldsymbol{x}_i'=(\alpha_1,\alpha_2,\alpha_3,0,0)^{\mathrm{T}}$ 为对应于 λ_i,$\boldsymbol{A}$ 的特征向量,

$\boldsymbol{y}_i'=(0,0,0,\beta_1,\beta_2)^{\mathrm{T}}$ 为对应于 λ_j,$\boldsymbol{A}$ 的特征向量.

第 9 章　常微分方程初值问题数值解法

9.1 引　　言

科学技术中常常需要求解常微分方程的定解问题.这类问题最简单的形式,是本章将要着重考察的一阶方程的初值问题

$$\begin{cases} y' = f(x,y), & (1.1) \\ y(x_0) = y_0. & (1.2) \end{cases}$$

我们知道,只要函数 $f(x,y)$ 适当光滑——譬如关于 y 满足利普希茨(Lipschitz)条件

$$| f(x,y) - f(x,\bar{y}) | \leqslant L | y - \bar{y} |. \tag{1.3}$$

理论上就可以保证初值问题(1.1),(1.2)的解 $y=y(x)$ 存在并且唯一.

虽然求解常微分方程有各种各样的解析方法,但解析方法只能用来求解一些特殊类型的方程,实际问题中归结出来的微分方程主要靠数值解法.

所谓**数值解法**,就是寻求解 $y(x)$ 在一系列离散节点

$$x_1 < x_2 < \cdots < x_n < x_{n+1} < \cdots$$

上的近似值 $y_1, y_2, \cdots, y_n, y_{n+1}, \cdots$.相邻两个节点的间距 $h_n = x_{n+1} - x_n$ 称为**步长**.今后如不特别说明,总是假定 $h_i = h(i=1,2,\cdots)$ 为定数,这时节点为 $x_n = x_0 + nh, n=0,1,2,\cdots$.

初值问题(1.1),(1.2)的数值解法有个基本特点,它们都采取"步进式",即求解过程顺着节点排列的次序一步一步地向前推进.描述这类算法,只要给出用已知信息 $y_n, y_{n-1}, y_{n-2}, \cdots$ 计算 y_{n+1} 的递推公式.

首先，要对方程(1.1)离散化，建立求数值解的递推公式．一类是计算 y_{n+1} 时只用到前一点的值 y_n，称为**单步法**．另一类是用到 y_{n+1} 前面 k 点的值 $y_n, y_{n-1}, \cdots, y_{n-k+1}$，称为 ***k* 步法**．其次，要研究公式的局部截断误差和阶，数值解 y_n 与精确解 $y(x_n)$ 的误差估计及收敛性，还有递推公式的计算稳定性等问题．

9.2 简单的数值方法与基本概念

9.2.1 欧拉法与后退欧拉法

我们知道，在 xy 平面上，微分方程(1.1)的解 $y=y(x)$ 称作它的**积分曲线**．积分曲线上一点 (x,y) 的切线斜率等于函数 $f(x,y)$ 的值．如果按函数 $f(x,y)$ 在 xy 平面上建立一个方向场，那么，积分曲线上每一点的切线方向均与方向场在该点的方向相一致．

基于上述几何解释，我们从初始点 $P_0(x_0,y_0)$ 出发，先依方向场在该点的方向推进到 $x=x_1$ 上一点 P_1，然后再从 P_1 依方向场的方向推进到 $x=x_2$ 上一点 P_2，循此前进做出一条折线 $\overline{P_0P_1P_2\cdots}$(图 9-1)．

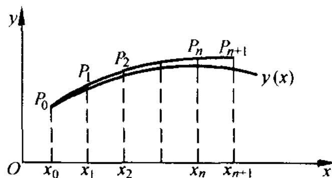


图 9-1

一般地，设已做出该折线的顶点 P_n，过 $P_n(x_n,y_n)$ 依方向场的方向再推进到 $P_{n+1}(x_{n+1},y_{n+1})$，显然两个顶点 P_n, P_{n+1} 的坐标有关系

$$\frac{y_{n+1}-y_n}{x_{n+1}-x_n}=f(x_n,y_n),$$

即

$$y_{n+1}=y_n+hf(x_n,y_n). \tag{2.1}$$

这就是著名的**欧拉(Euler)公式**．若初值 y_0 已知，则依公式(2.1)可逐步算出

$$y_1 = y_0 + hf(x_0, y_0),$$
$$y_2 = y_1 + hf(x_1, y_1),$$
$$\cdots$$

例 1　求解初值问题

$$\begin{cases} y' = y - \dfrac{2x}{y} \quad (0 < x < 1), \\ y(0) = 1. \end{cases} \tag{2.2}$$

解　为便于进行比较，本章将用多种数值方法求解上述初值问题. 这里先用欧拉方法，欧拉公式的具体形式为

$$y_{n+1} = y_n + h\left(y_n - \frac{2x_n}{y_n}\right).$$

取步长 $y=0.1$，计算结果见表 9-1.

表 9-1　计算结果对比

x_n	y_n	$y(x_n)$	x_n	y_n	$y(x_n)$
0.1	1.1000	1.0954	0.6	1.5090	1.4832
0.2	1.1918	1.1832	0.7	1.5803	1.5492
0.3	1.2774	1.2649	0.8	1.6498	1.6125
0.4	1.3582	1.3416	0.9	1.7178	1.6733
0.5	1.4351	1.4142	1.0	1.7848	1.7321

初值问题(2.2)有解 $y=\sqrt{1+2x}$，按这个解析式子算出的准确值 $y(x_n)$ 同近似值 y_n 一起列在表 9-1 中，两者相比较可以看出欧拉方法的精度很差.

还可以通过几何直观来考察欧拉方法的精度. 假设 $y_n=y(x_n)$，即顶点 P_n 落在积分曲线 $y=y(x)$ 上，那么，按欧拉方法做出的折线 P_nP_{n+1} 便是 $y=y(x)$ 过点 P_n 的切线(图 9-2). 从图形上看，这样定

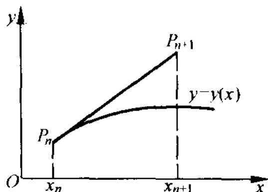


图　9-2

出的顶点 P_{n+1} 显著地偏离了原来的积分曲线，可见欧拉方法是相当粗糙的.

为了分析计算公式的精度，通常可用泰勒展开将 $y(x_{n+1})$ 在 x_n 处展开，则有

$$
\begin{aligned}
y(x_{n+1}) &= y(x_n + h) \\
&= y(x_n) + y'(x_n)h + \frac{h^2}{2}y''(\xi_n), \quad \xi_n \in (x_n, x_{n+1}).
\end{aligned}
$$

在 $y_n = y(x_n)$ 的前提下，$f(x_n, y_n) = f(x_n, y(x_n)) = y'(x_n)$. 于是可得欧拉法(2.1)的公式误差

$$
y(x_{n+1}) - y_{n+1} = \frac{h^2}{2}y''(\xi_n) \approx \frac{h^2}{2}y''(x_n), \tag{2.3}
$$

称为此方法的局部截断误差.

如果对方程(1.1)从 x_n 到 x_{n+1} 积分，得

$$
y(x_{n+1}) = y(x_n) + \int_{x_n}^{x_{n+1}} f(t, y(t))\mathrm{d}t. \tag{2.4}
$$

右端积分用左矩形公式 $hf(x_n, y(x_n))$ 近似，再以 y_n 代替 $y(x_n)$，y_{n+1} 代替 $y(x_{n+1})$ 也得到(2.1)，局部截断误差也是(2.3).

如果在(2.4)中右端积分用右矩形公式 $hf(x_{n+1}, y(x_{n+1}))$ 近似，则得另一个公式

$$
y_{n+1} = y_n + hf(x_{n+1}, y_{n+1}), \tag{2.5}
$$

称为**后退的欧拉法**.

后退的欧拉公式与欧拉公式有着本质的区别，后者是关于 y_{n+1} 的一个直接的计算公式，这类公式称作是**显式的**；然而公式(2.5)的右端含有未知的 y_{n+1}，它实际上是关于 y_{n+1} 的一个函数方程，这类公式称作是**隐式的**.

显式与隐式两类方法各有特点. 考虑到数值稳定性等其他因素，人们有时需要选用隐式方法，但使用显式算法远比隐式方便.

隐式方程(2.5)通常用迭代法求解，而迭代过程的实质是逐步显示化.

设用欧拉公式

$$y_{n+1}^{(0)} = y_n + hf(x_n, y_n)$$

给出迭代初值 $y_{n+1}^{(0)}$，用它代入(2.5)式的右端，使之转化为显式，直接计算得

$$y_{n+1}^{(1)} = y_n + hf(x_{n+1}, y_{n+1}^{(0)}),$$

然后再用 $y_{n+1}^{(1)}$ 代入(2.5)式，又有

$$y_{n+1}^{(2)} = y_n + hf(x_{n+1}, y_{n+1}^{(1)}).$$

如此反复进行，得

$$y_{n+1}^{(k+1)} = y_n + hf(x_{n+1}, y_{n+1}^{(k)}) \quad (k = 0,1,\cdots). \tag{2.6}$$

由于 $f(x,y)$ 对 y 满足利普希茨条件(1.3). 由(2.6)减(2.5)得

$$\begin{aligned} |y_{n+1}^{(k+1)} - y_{n+1}| &= h|f(x_{n+1}, y_{n+1}^{(k)}) - f(x_{n+1}, y_{n+1})| \\ &\leqslant hL|y_{n+1}^{(k)} - y_{n+1}|. \end{aligned}$$

由此可知，只要 $hL<1$ 迭代法(2.6)就收敛到解 y_{n+1}. 关于后退欧拉方法的公式误差，从积分公式看到它与欧拉法是相似的.

9.2.2 梯形方法

为得到比欧拉法精度高的计算公式，在等式(2.4)右端积分中若用梯形求积公式近似，并用 y_n 代替 $y(x_n)$，y_{n+1} 代替 $y(x_{n+1})$，则得

$$y_{n+1} = y_n + \frac{h}{2}[f(x_n, y_n) + f(x_{n+1}, y_{n+1})], \tag{2.7}$$

称为**梯形方法**.

梯形方法是隐式单步法，可用迭代法求解. 同后退的欧拉方法一样，仍用欧拉方法提供迭代初值，则梯形法的迭代公式为

$$\begin{cases} y_{n+1}^{(0)} = y_n + hf(x_n, y_n); \\ y_{n+1}^{(k+1)} = y_n + \dfrac{h}{2}[f(x_n, y_n) + f(x_{n+1}, y_{n+1}^{(k)})] \\ (k = 0,1,2,\cdots). \end{cases} \tag{2.8}$$

为了分析迭代过程的收敛性,将(2.7)式与(2.8)式相减,得

$$y_{n+1} - y_{n+1}^{(k+1)} = \frac{h}{2}[f(x_{n+1}, y_{n+1}) - f(x_{n+1}, y_{n+1}^{(k)})],$$

于是有

$$| y_{n+1} - y_{n+1}^{(k+1)} | \leqslant \frac{hL}{2} | y_{n+1} - y_{n+1}^{(k)} |,$$

式中 L 为 $f(x,y)$ 关于 y 的利普希茨常数.如果选取 h 充分小,使得

$$\frac{hL}{2} < 1,$$

则当 $k \to \infty$ 时有 $y_{n+1}^{(k)} \to y_{n+1}$,这说明迭代过程(2.8)是收敛的.

9.2.3 单步法的局部截断误差与阶

初值问题(1.1),(1.2)的单步法可用一般形式表示为

$$y_{n+1} = y_n + h\varphi(x_n, y_n, y_{n+1}, h), \tag{2.9}$$

其中多元函数 φ 与 $f(x,y)$ 有关,当 φ 含有 y_{n+1} 时,方法是隐式的,若不含 y_{n+1} 则为显式方法,所以显式单步法可表示为

$$y_{n+1} = y_n + h\varphi(x_n, y_n, h), \tag{2.10}$$

$\varphi(x,y,h)$ 称为增量函数,例如对欧拉法(2.1)有

$$\varphi(x,y,h) = f(x,y).$$

它的局部截断误差已由(2.3)给出,对一般显式单步法则可如下定义.

定义 1 设 $y(x)$ 是初值问题(1.1),(1.2)的准确解,称

$$T_{n+1} = y(x_{n+1}) - y(x_n) - h\varphi(x_n, y(x_n), h) \tag{2.11}$$

为显式单步法(2.10)的**局部截断误差**.

T_{n+1} 之所以称为局部的,是假设在 x_n 前各步没有误差.当 $y_n = y(x_n)$ 时,计算一步,则有

$$\begin{aligned} y(x_{n+1}) - y_{n+1} &= y(x_{n+1}) - [y_n + h\varphi(x_n, y_n, h)] \\ &= y(x_{n+1}) - y(x_n) - h\varphi(x_n, y(x_n), h) = T_{n+1}. \end{aligned}$$

所以,局部截断误差可理解为用方法(2.10)计算一步的误差,也即公式(2.10)中用准确解 $y(x)$ 代替数值解产生的公式误差.根据定义,显然欧拉法的局部截断误差

$$\begin{aligned} T_{n+1} &= y(x_{n+1}) - y(x_n) - hf(x_n, y(x_n)) \\ &= y(x_n + h) - y(x_n) - hy'(x_n) \\ &= \frac{h^2}{2} y''(x_n) + O(h^3), \end{aligned}$$

即为(2.3)的结果.这里 $\frac{h^2}{2} y''(x_n)$ 称为局部截断误差主项.显然 $T_{n+1} = O(h^2)$.一般情形的定义如下.

定义 2 设 $y(x)$ 是初值问题(1.1),(1.2)的准确解,若存在最大整数 p 使显式单步法(2.10)的局部截断误差满足

$$T_{n+1} = y(x+h) - y(x) - h\varphi(x, y, h) = O(h^{p+1}), \tag{2.12}$$

则称方法(2.10)具有 **p 阶精度**.

若将(2.12)展开式写成

$$T_{n+1} = \psi(x_n, y(x_n)) h^{p+1} + O(h^{p+2}),$$

则 $\psi(x_n, y(x_n)) h^{p+1}$ 称为**局部截断误差主项**.

以上定义对隐式单步法(2.9)也是适用的.例如,对后退欧拉法(2.5)其局部截断误差为

$$\begin{aligned} T_{n+1} &= y(x_{n+1}) - y(x_n) - hf(x_{n+1}, y(x_{n+1})) \\ &= hy'(x_n) + \frac{h^2}{2} y''(x_n) + O(h^3) \\ &\quad - h[y'(x_n) + hy''(x_n) + O(h^2)] \\ &= -\frac{h^2}{2} y''(x_n) + O(h^3). \end{aligned}$$

这里 $p=1$,是 1 阶方法,局部截断误差主项为 $-\frac{h^2}{2} y''(x_n)$.

同样对梯形法(2.7)有

$$
\begin{aligned}
T_{n+1} &= y(x_{n+1}) - y(x_n) - \frac{h}{2}[y'(x_n) + y'(x_{n+1})] \\
&= hy'(x_n) + \frac{h^2}{2}y''(x_n) + \frac{h^3}{3!}y'''(x_n) \\
&\quad - \frac{h}{2}[y'(x_n) + y'(x_n) + hy''(x_n) + \frac{h^2}{2}y'''(x_n)] + O(h^4) \\
&= -\frac{h^3}{12}y'''(x_n) + O(h^4).
\end{aligned}
$$

所以梯形方法(2.7)是二阶的，其局部误差主项为$-\frac{h^3}{12}y'''(x_n)$.

9.2.4　改进的欧拉公式

我们看到，梯形方法虽然提高了精度，但其算法复杂，在应用迭代公式(2.9)进行实际计算时，每迭代一次，都要重新计算函数$f(x,y)$的值，而迭代又要反复进行若干次，计算量很大，而且往往难以预测. 为了控制计算量，通常只迭代一两次就转入下一步的计算，这就简化了算法.

具体地说，我们先用欧拉公式求得一个初步的近似值$\bar{y}_{n+1}$，称之为**预测值**，预测值$\bar{y}_{n+1}$的精度可能很差，再用梯形公式(2.7)将它校正一次，即按(2.8)式迭代一次得y_{n+1}，这个结果称**校正值**，而这样建立的预测-校正系统通常称为**改进的欧拉公式**：

$$
\begin{aligned}
&\text{预测}\quad \bar{y}_{n+1} = y_n + hf(x_n, y_n), \\
&\text{校正}\quad y_{n+1} = y_n + \frac{h}{2}[f(x_n, y_n) + f(x_{n+1}, \bar{y}_{n+1})].
\end{aligned}
\tag{2.13}
$$

或表为下列平均化形式

$$
\begin{cases}
y_p = y_n + hf(x_n, y_n), \\
y_c = y_n + hf(x_{n+1}, y_p), \\
y_{n+1} = \frac{1}{2}(y_p + y_c).
\end{cases}
$$

例 2　用改进的欧拉方法求解初值问题(2.2).

解 改进的欧拉公式为

$$\begin{cases} y_p = y_n + h\left(y_n - \dfrac{2x_n}{y_n}\right), \\ y_c = y_n + h\left(y_p - \dfrac{2x_{n+1}}{y_p}\right), \\ y_{n+1} = \dfrac{1}{2}(y_p + y_c). \end{cases}$$

仍取 $h=0.1$，计算结果见表 9-2. 同例 1 中欧拉法的计算结果比较，改进欧拉法明显改善了精度.

表 9-2 计算结果对比

x_n	y_n	$y(x_n)$	x_n	y_n	$y(x_n)$
0.1	1.0959	1.0954	0.6	1.4860	1.4832
0.2	1.1841	1.1832	0.7	1.5525	1.5492
0.3	1.2662	1.2649	0.8	1.6153	1.6165
0.4	1.3434	1.3416	0.9	1.6782	1.6733
0.5	1.4164	1.4142	1.0	1.7379	1.7321

9.3 龙格-库塔方法

9.3.1 显式龙格-库塔法的一般形式

上节给出了显式单步法的表达式(2.10)，其局部截断误差为(2.12)，对欧拉法 $T_{n+1}=O(h^2)$，即方法为 $p=1$ 阶，若用改进欧拉法(2.13)，它可表为

$$y_{n+1} = y_n + \frac{h}{2}[f(x_n, y_n) + f(x_n + h, y_n + hf(x_n, y_n))]. \tag{3.1}$$

此时增量函数

$$\varphi(x_n,y_n,h)=\frac{1}{2}[f(x_n,y_n)+f(x_n+h,y_n+hf(x_n,y_n))].\tag{3.2}$$

它比欧拉法的 $\varphi(x_n,y_n,h)=f(x_n,y_n)$，增加了计算一个右函数 f 的值，可望 $p=2$. 若要使得到的公式阶数 p 更大，φ 就必须包含更多的 f 值. 实际上从方程(1.1)等价的积分形式(2.4)，即

$$y(x_{n+1})-y(x_n)=\int_{x_n}^{x_{n+1}}f(x,y(x))\,\mathrm{d}x,\tag{3.3}$$

若要使公式阶数提高，就必须使右端积分的数值求积公式精度提高，它必然要增加求积节点，为此可将(3.3)的右端用求积公式表示为

$$\int_{x_n}^{x_{n+1}}f(x,y(x))\mathrm{d}x\approx h\sum_{i=1}^{r}c_if(x_n+\lambda_ih,y(x_n+\lambda_ih)).$$

一般来说，点数 r 越多，精度越高，上式右端相当于增量函数 $\varphi(x,y,h)$，为得到便于计算的显式方法，可类似于改进欧拉法(3.1)，(3.2)，将公式表示为

$$y_{n+1}=y_n+h\varphi(x_n,y_n,h),\tag{3.4}$$

其中

$$\varphi(x_n,y_n,h)=\sum_{i=1}^{r}c_iK_i,\tag{3.5}$$

$$K_1=f(x_n,y_n),$$

$$K_i=f(x_n+\lambda_ih,y_n+h\sum_{j=1}^{i-1}\mu_{ij}K_j)\qquad i=2,\cdots,r,$$

这里 c_i,λ_i,μ_{ij} 均为常数. (3.4)和(3.5)称为 r 级显式**龙格-库塔(Runge-Kutta)法**，简称 R-K 方法.

当 $r=1,\varphi(x_n,y_n,h)=f(x_n,y_n)$ 时，就是欧拉法，此时方法的阶为 $p=1$. 当 $r=2$ 时，改进欧拉法(3.1)，(3.2)就是其中的一种，下面将证明阶 $p=2$. 要使公式(3.4)，(3.5)具有更高的阶 p，就要增加点数 r. 下面我们只就 $r=2$ 推导 R-K 方法. 并给出 $r=3,4$ 时

的常用公式，其推导方法与 $r=2$ 时类似，只是计算较复杂.

9.3.2 二阶显式 R-K 方法

对 $r=2$ 的 R-K 方法，由(3.4)，(3.5)可得到如下的计算公式

$$\begin{cases} y_{n+1} = y_n + h(c_1 K_1 + c_2 K_2), \\ K_1 = f(x_n, y_n), \\ K_2 = f(x_n + \lambda_2 h,\ y_n + \mu_{21} h K_1). \end{cases} \tag{3.6}$$

这里 $c_1, c_2, \lambda_2, \mu_{21}$ 均为待定常数，我们希望适当选取这些系数，使公式阶数 p 尽量高. 根据局部截断误差定义，(3.6)的局部截断误差为

$$\begin{aligned} T_{n+1} = &y(x_{n+1}) - y(x_n) \\ &- h[c_1 f(x_n, y_n) + c_2 f(x_n + \lambda_2 h, y_n + \mu_{21} h f_n)], \end{aligned} \tag{3.7}$$

这里 $y_n = y(x_n)$，$f_n = f(x_n, y_n)$. 为得到 T_{n+1} 的阶 p，要将上式各项在(x_n, y_n)处做泰勒展开，由于 $f(x, y)$是二元函数，故要用到二元泰勒展开，各项展开式为

$$y(x_{n+1}) = y_n + h y_n' + \frac{h^2}{2} y_n'' + \frac{h^3}{3!} y_n''' + O(h^4),$$

其中

$$\begin{cases} y_n' = f(x_n, y_n) = f_n, \\ y_n'' = \dfrac{\mathrm{d}}{\mathrm{d}x} f(x_n, y(x_n)) = f_x'(x_n, y_n) + f_y'(x_n, y_n) \cdot f_n, \\ y_n''' = f_{xx}''(x_n, y_n) + 2 f_n f_{xy}'(x_n, y_n) + f_n^2 f_{yy}''(x_n, y_n) \\ \qquad\quad + f_y'(x_n, y_n)[f_x'(x_n, y_n) + f_n f_y'(x_n, y_n)]; \end{cases} \tag{3.8}$$

$$\begin{aligned} &f(x_n + \lambda_2 h, y_n + \mu_{21} h f_n) \\ &\quad = f_n + f_x'(x_n, y_n) \lambda_2 h + f_y'(x_n, y_n) \mu_{21} h f_n + O(h^2). \end{aligned}$$

将以上结果代入(3.7)则有

$$T_{n+1} = h f_n + \frac{h^2}{2}[f_x'(x_n, y_n) + f_y'(x_n, y_n) f_n]$$

$$
\begin{aligned}
&-h[c_1 f_n + c_2(f_n + \lambda_2 f_x'(x_n, y_n)h \\
&+ \mu_{21} f_y'(x_n, y_n) f_n h] + O(h^3) \\
= &(1 - c_1 - c_2) f_n h + \left(\frac{1}{2} - c_2\lambda_2\right) f_x'(x_n, y_n) h^2 \\
&+ \left(\frac{1}{2} - c_2\mu_{21}\right) f_y'(x_n, y_n) f_n h^2 + O(h^3).
\end{aligned}
$$

要使公式(3.6)具有 $p=2$ 阶，必须使

$$
1 - c_1 - c_2 = 0,\ \frac{1}{2} - c_2\lambda_2 = 0,\ \frac{1}{2} - c_2\mu_{21} = 0. \tag{3.9}
$$

即

$$
c_2\lambda_2 = \frac{1}{2},\ c_2\mu_{21} = \frac{1}{2},\ c_1 + c_2 = 1.
$$

(3.9)的解是不唯一的. 可令 $c_2 = a \neq 0$，则得

$$
c_1 = 1 - a,\ \lambda_2 = \mu_{21} = \frac{1}{2a}.
$$

这样得到的公式称为二阶 R-K 方法，如取 $a = 1/2$，则 $c_1 = c_2 = 1/2$，$\lambda_2 = \mu_{21} = 1$. 这就是改进欧拉法(3.1).

若取 $a=1$，则 $c_2 = 1, c_1 = 0, \lambda_2 = \mu_{21} = 1/2$. 得计算公式

$$
\begin{cases}
y_{n+1} = y_n + hK_2, \\
K_1 = f(x_n, y_n), \\
K_2 = f\left(x_n + \dfrac{h}{2}, y_n + \dfrac{h}{2}K_1\right).
\end{cases} \tag{3.10}
$$

称为**中点公式**，相当于数值积分的中矩形公式. (3.10)也可表示为

$$
y_{n+1} = y_n + hf\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}f(x_n, y_n)\right).
$$

对 $r=2$ 的 R-K 公式(3.6)能否使局部误差提高到 $O(h^4)$？为此需把 K_2 多展开一项，从(3.8)的 y_n''' 看到展开式中 $f_y' f_x' + f_y' f$ 的项是不能通过选择参数消掉的，实际上要使 h^3 的项为零，需增加 3 个方程，要确定 4 个参数 c_1, c_2, λ_2 及 μ_{21}，这是不可能的. 故 $r=2$ 的显式 R-K 方法的阶只能是 $p=2$，而不能得到三阶公式.

9.3.3　三阶与四阶显式 R-K 方法

要得到三阶显式 R-K 方法，必须 $r=3$. 此时(3.4)，(3.5)的公式表示为

$$\begin{cases} y_{n+1} = y_n + h(c_1K_1 + c_2K_2 + c_3K_3), \\ K_1 = f(x_n, y_n), \\ K_2 = f(x_n + \lambda_2 h, y_n + \mu_{21}hK_1), \\ K_3 = f(x_n + \lambda_3 h, y_n + \mu_{31}hK_1 + \mu_{32}hK_2). \end{cases} \tag{3.11}$$

其中 c_1, c_2, c_3 及 $\lambda_2, \mu_{21}, \lambda_3, \mu_{31}, \mu_{32}$ 均为待定参数，公式(3.11)的局部截断误差为

$$T_{n+1} = y(x_{n+1}) - y(x_n) - h[c_1K_1 + c_2K_2 + c_3K_3].$$

只要将 K_2, K_3 按二元函数泰勒展开，使 $T_{n+1}=O(h^4)$，可得待定参数满足方程

$$\begin{cases} c_1 + c_2 + c_3 = 1, \\ \lambda_2 = \mu_{21}, \\ \lambda_3 = \mu_{31} + \mu_{32}, \\ c_2\lambda_2 + c_3\lambda_3 = \dfrac{1}{2}, \\ c_2\lambda_2^2 + c_3\lambda_3^2 = \dfrac{1}{3}, \\ c_3\lambda_2\mu_{32} = \dfrac{1}{6}. \end{cases} \tag{3.12}$$

这是 8 个未知数 6 个方程的方程组，解也不是唯一的. 可以得到很多公式. 满足条件(3.12)的公式(3.11)统称为三阶 R-K 公式. 下面只给出其中一个常见的公式.

$$
\begin{cases}
y_{n+1} = y_n + \dfrac{h}{6}(K_1 + 4K_2 + K_3), \\
K_1 = f(x_n, y_n), \\
K_2 = f\left(x_n + \dfrac{h}{2}, y_n + \dfrac{h}{2}K_1\right), \\
K_3 = f(x_n + h, y_n - hK_1 + 2hK_2).
\end{cases}
$$

此公式称为**库塔**三阶方法.

继续上述过程,经过较复杂的数学演算,可以导出各种四阶龙格-库塔公式,下列经典公式是其中常用的一个:

$$
\begin{cases}
y_{n+1} = y_n + \dfrac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4), \\
K_1 = f(x_n, y_n), \\
K_2 = f\left(x_n + \dfrac{h}{2}, y_n + \dfrac{h}{2}K_1\right), \\
K_3 = f\left(x_n + \dfrac{h}{2}, y_n + \dfrac{h}{2}K_2\right), \\
K_4 = f(x_n + h, y_n + hK_3).
\end{cases}
\tag{3.13}
$$

四阶龙格-库塔方法的每一步需要计算四次函数值 f,可以证明其截断误差为 $O(h^5)$. 不过证明极其繁琐,这里从略.

例 3 设取步长 $h=0.2$,从 $x=0$ 直到 $x=1$ 用四阶龙格-库塔方法求解初值问题(2.2).

解 这里,经典的四阶龙格-库塔公式(3.13)具有形式

$$
\begin{cases}
y_{n+1} = y_n + \dfrac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4), \\
K_1 = y_n - \dfrac{2x_n}{y_n}, \\
K_2 = y_n + \dfrac{h}{2}K_1 - \dfrac{2x_n + h}{y_n + \dfrac{h}{2}K_1}, \\
K_3 = y_n + \dfrac{h}{2}K_2 - \dfrac{2x_n + h}{y_n + \dfrac{h}{2}K_2}, \\
K_4 = y_n + hK_3 - \dfrac{2(x_n + h)}{y_n + hK_3}.
\end{cases}
$$

右面列出计算结果 y_n，表中 $y(x_n)$ 仍表示准确解.

比较例 3 和例 2 的计算结果，显然以龙格-库塔方法的精度为高. 要注意，虽然四阶龙格-库塔方法的计算量（每一步要 4 次计算函数 f）比改进的欧拉方法（它是一种二阶龙格-库塔方法，每一步只要 2 次计算函数 f）大一倍，但由于这里放大了步长（$h=0.2$），表 9-3 和表 9-2 所耗费的计算量几乎相同. 这个例子又一次显示了选择算法的重要意义.

表 9-3 计算结果

x_n	y_n	$y(x_n)$
0.2	1.1832	1.1832
0.4	1.3417	1.3416
0.6	1.4833	1.4832
0.8	1.6125	1.6125
1.0	1.7321	1.7321

然而值得指出的是，龙格-库塔方法的推导基于泰勒展开方法，因而它要求所求的解具有较好的光滑性质. 反之，如果解的光滑性差，那么，使用四阶龙格-库塔方法求得的数值解，其精度可能反而不如改进的欧拉方法. 实际计算时，我们应当针对问题的具体特点选择合适的算法.

9.3.4 变步长的龙格-库塔方法

单从每一步看，步长越小，截断误差就越小，但随着步长的缩小，在一定求解范围内所要完成的步数就增加了.步数的增加不但引起计算量的增大，而且可能导致舍入误差的严重积累.因此同积分的数值计算一样，微分方程的数值解法也有个选择步长的问题.

在选择步长时，需要考虑两个问题：

1° 怎样衡量和检验计算结果的精度？

2° 如何依据所获得的精度处理步长？

我们考察经典的四阶龙格-库塔公式(3.13).从节点 x_n 出发，先以 h 为步长求出一个近似值，记为 $y_{n+1}^{(h)}$，由于公式的局部截断误差为 $O(h^5)$，故有

$$y(x_{n+1})-y_{n+1}^{(h)}\approx ch^5, \tag{3.14}$$

然后将步长折半，即取 $\frac{h}{2}$ 为步长从 x_n 跨两步到 x_{n+1}，再求得一个近似值 $y_{n+1}^{(\frac{h}{2})}$，每跨一步的截断误差是 $c\left(\frac{h}{2}\right)^5$，因此有

$$y(x_{n+1})-y_{n+1}^{(\frac{h}{2})}\approx 2c\left(\frac{h}{2}\right)^5, \tag{3.15}$$

比较(3.14)式和(3.15)式我们看到，步长折半后，误差大约减少到 $\frac{1}{16}$，即有

$$\frac{y(x_{n+1})-y_{n+1}^{(\frac{h}{2})}}{y(x_{n+1})-y_{n+1}^{(h)}}\approx\frac{1}{16}.$$

由此易得下列事后估计式

$$y(x_{n+1})-y_{n+1}^{(\frac{h}{2})}\approx\frac{1}{15}\left[y_{n+1}^{(\frac{h}{2})}-y_{n+1}^{(h)}\right].$$

这样，我们可以通过检查步长，折半前后两次计算结果的偏差

$$\Delta=\left|y_{n+1}^{(\frac{h}{2})}-y_{n+1}^{(h)}\right|$$

来判定所选的步长是否合适，具体地说，将区分以下两种情况处理：

1. 对于给定的精度 ε，如果 $\Delta>\varepsilon$，我们反复将步长折半进行计算，直至 $\Delta<\varepsilon$ 为止，这时取最终得到的 $y_{n+1}^{(\frac{h}{2})}$ 作为结果；

2. 如果 $\Delta<\varepsilon$，我们将反复将步长加倍，直到 $\Delta>\varepsilon$ 为止，这时再将步长折半一次，就得到所要的结果.

这种通过加倍或折半处理步长的方法称为**变步长方法**. 表面上看，为了选择步长，每一步的计算量增加了，但总体考虑往往是合算的.

9.4　单步法的收敛性与稳定性

9.4.1　收敛性与相容性

数值解法的基本思想是，通过某种离散化手段将微分方程(1.1)转化为差分方程，如单步法(2.10)，即

$$y_{n+1}=y_n+h\varphi(x_n,y_n,h). \tag{4.1}$$

它在 x_n 处的解为 y_n，而初值问题(1.1)，(1.2)在 x_n 处的精确解为 $y(x_n)$，记 $e_n=y(x_n)-y_n$ 称为整体截断误差. 收敛性就是讨论当 $x=x_n$ 固定且 $h=\frac{x_n-x_0}{n}\to 0$ 时 $e_n\to 0$ 的问题.

定义 3　若一种数值方法(如单步法(4.1))对于固定的 $x_n=x_0+nh$，当 $h\to 0$ 时有 $y_n\to y(x_n)$，其中 $y(x)$ 是(1.1)，(1.2)的准确解，则称该方法是**收敛**的.

显然数值方法收敛是指 $e_n=y(x_n)-y_n\to 0$，对单步法(4.1)有下述收敛性定理：

定理 1　假设单步法(4.1)具有 p 阶精度，且增量函数 $\varphi(x,y,h)$ 关于 y 满足利普希茨条件

$$|\varphi(x,y,h)-\varphi(x,\bar{y},h)|\leqslant L_{\varphi}|y-\bar{y}|, \tag{4.2}$$

又设初值 y_0 是准确的，即 $y_0=y(x_0)$，则其**整体截断误差**

$$y(x_n)-y_n=O(h^p). \tag{4.3}$$

证明 设以 $\bar{y}_{n+1}$ 表示取 $y_n=y(x_n)$ 用公式(4.1)求得的结果，即

$$\bar{y}_{n+1}=y(x_n)+h\varphi(x_n,y(x_n),h), \tag{4.4}$$

则 $y(x_{n+1})-\bar{y}_{n+1}$ 为局部截断误差，由于所给方法具有 p 阶精度，按定义 2，存在定数 C，使

$$|y(x_{n+1})-\bar{y}_{n+1}|\leqslant Ch^{p+1}.$$

又由式(4.4)与(4.1)，得

$$|\bar{y}_{n+1}-y_{n+1}|\leqslant|y(x_n)-y_n|+h|\varphi(x_n,y(x_n),h)-\varphi(x_n,y_n,h)|.$$

利用假设条件(4.2)，有

$$|\bar{y}_{n+1}-y_{n+1}|\leqslant(1+hL_{\varphi})|y(x_n)-y_n|,$$

从而有

$$\begin{aligned}|y(x_{n+1})-y_{n+1}|&\leqslant|\bar{y}_{n+1}-y_{n+1}|+|y(x_{n+1})-\bar{y}_{n+1}|\\&\leqslant(1+hL_{\varphi})|y(x_n)-y_n|+Ch^{p+1}.\end{aligned}$$

即对整体截断误差 $e_n=y(x_n)-y_n$ 成立下列递推关系式

$$|e_{n+1}|\leqslant(1+hL_{\varphi})|e_n|+Ch^{p+1}, \tag{4.5}$$

据此不等式反复递推，可得

$$|e_n|\leqslant(1+hL_{\varphi})^n|e_0|+\frac{Ch^p}{L_{\varphi}}[(1+hL_{\varphi})^n-1]. \tag{4.6}$$

再注意到当 $x_n-x_0=nh\leqslant T$ 时

$$(1+hL_{\varphi})^n\leqslant(e^{hL_{\varphi}})^n\leqslant e^{TL_{\varphi}},$$

最终得下列估计式

[注] 对于任意实数 x，有 $1+x\leqslant e^x$，而当 $x\geqslant-1$ 时，成立 $0\leqslant(1+x)^n\leqslant e^{nx}$.

$$|e_n| \leqslant |e_0| \mathrm{e}^{TL_\varphi} + \frac{Ch^p}{L_\varphi}(\mathrm{e}^{TL_\varphi} - 1). \tag{4.7}$$

由此可以断定，如果初值是准确的，即 $e_0=0$，则(4.3)式成立. 定理证毕.

依据这一定理，判断单步法(4.1)的收敛性，归结为验证增量函数 φ 能否满足利普希茨条件(4.2).

对于欧拉方法，由于其增量函数 φ 就是 $f(x,y)$，故当 $f(x,y)$ 关于 y 满足利普希茨条件时它是收敛的.

再考察改进的欧拉方法，其增量函数已由(3.2)式给出，这时有

$$|\varphi(x,y,h) - \varphi(x,\bar{y},h)| \leqslant \frac{1}{2}[|f(x,y) - f(x,\bar{y})|$$
$$+|f(x+h,y+hf(x,y)) - f(x+h,\bar{y}+hf(x,\bar{y}))|].$$

假设 $f(x,y)$ 关于 y 满足利普希茨条件，记利普希茨常数为 L，则由上式推得

$$|\varphi(x,y,h) - \varphi(x,\bar{y},h)| \leqslant L\left(1+\frac{h}{2}L\right)|y-\bar{y}|.$$

设限定 $h\leqslant h_0$（h_0 为定数），上式表明 φ 关于 y 的利普希茨常数

$$L_\varphi = L\left(1+\frac{h_0}{2}L\right),$$

因此改进的欧拉方法也是收敛的.

类似地，不难验证其他龙格-库塔方法的收敛性.

定理 1 表明 $p\geqslant 1$ 时单步法收敛，并且当 $y(x)$ 是初值问题(1.1)，(1.2)的解，(4.1)具有 p 阶精度时，则有展开式

$$\begin{aligned} T_{n+1} &= y(x+h) - y(x) - h\varphi(x,y(x),h) \\ &= y'(x)h + \frac{y''(x)}{2}h^2 + \cdots \\ &\quad - h[\varphi(x,y(x),0) + \varphi_x'(x,y(x),0)h + \cdots] \\ &= h[y'(x) - \varphi(x,y(x),0)] + O(h^2). \end{aligned}$$

所以 $p \geqslant 1$ 的充要条件是 $y'(x)-\varphi(x,y(x),0)=0$，而 $y'(x)=f(x,y(x))$，于是可给出如下定义：

定义 4 若单步法(4.1)的增量函数 φ 满足

$$\varphi(x,y,0) = f(x,y),$$

则称单步法(4.1)与初值问题(1.1)，(1.2)**相容**.

以上讨论表明 p 阶方法(4.1)当 $p \geqslant 1$ 时与(1.1)，(1.2)相容，反之相容方法至少是 1 阶的.

于是由定理 1 可知方法(4.1)收敛的充分必要条件是此方法是相容的.

9.4.2 绝对稳定性与绝对稳定域

前面关于收敛性的讨论有个前提，必须假定数值方法本身的计算是准确的. 实际情形并不是这样，差分方程的求解还会有计算误差，譬如由于数字舍入而引起的小扰动. 这类小扰动在传播过程中会不会恶性增长，以至于"淹没"了差分方程的"真解"呢? 这就是差分方法的稳定性问题. 在实际计算时，我们希望某一步产生的扰动值，在后面的计算中能够被控制，甚至是逐步衰减的.

定义 5 若一种数值方法在节点值 y_n 上大小为 δ 的扰动，于以后各节点值 $y_m(m>n)$ 上产生的偏差均不超过 δ，则称该方法是**稳定**的.

下面先以欧拉法为例考察计算稳定性.

例 4 考察初值问题

$$\begin{cases} y' = -100y, \\ y(0) = 1. \end{cases}$$

其准确解 $y(x)=\mathrm{e}^{-100x}$ 是一个按指数曲线衰减得很快的函数，如图 9-3 所示.

用欧拉法解方程 $y'=-100y$ 得

$$y_{n+1} = (1-100h)y_n.$$

若取 $h=0.025$，则欧拉公式的具体形式为

$$y_{n+1}=-1.5y_n,$$

计算结果列于表 9-4 的第 2 列. 我们看到，欧拉方法的解 y_n（图 9-3 中用×号标出）在准确值 $y(x_n)$ 的上下波动，计算过程明显地不稳定. 但若取 $h=0.005$，$y_{n+1}=0.5y_n$ 则计算过程稳定.

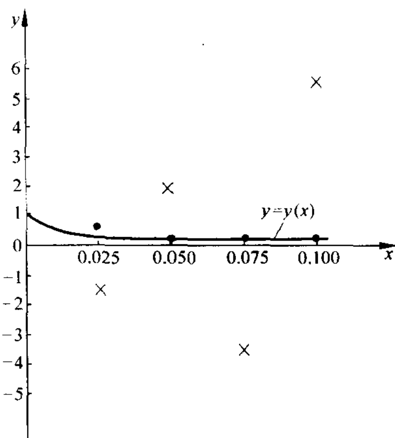


图 9-3

再考察后退的欧拉方法，取 $h=0.025$ 时计算公式为

$$y_{n+1}=\frac{1}{3.5}y_n.$$

计算结果列于表 9-4 的第 3 列（图 9-3 中标以·号），这时计算过程是稳定的.

表 9-4 计算结果对比

节　　点	欧拉方法	后退欧拉方法
0.025	−1.5	0.2857
0.050	2.25	0.0816
0.075	−3.375	0.0233
0.100	5.0625	0.0067

例题表明稳定性不但与方法有关，也与步长 h 的大小有关，当然也与方程中的 $f(x,y)$ 有关. 为了只考察数值方法本身. 通常只检验将数值方法用于解模型方程的稳定性，模型方程为

$$y'=\lambda y, \tag{4.8}$$

其中 λ 为复数，这个方程分析较简单. 对一般方程可以通过局部线性化化为这种形式，例如在 $(\bar{x},\bar{y})$ 的邻域，可展开为

$$\begin{aligned} y' &= f(x,y) \\ &= f(\bar{x},\bar{y}) + f_x'(\bar{x},\bar{y})(x-\bar{x}) + f_y'(\bar{x},\bar{y})(y-\bar{y}) + \cdots, \end{aligned}$$

略去高阶项，再做变换即可得到 $u'=\lambda u$ 的形式. 对于 m 个方程的方程组，可线性化为 $\boldsymbol{y}'=\boldsymbol{A}\boldsymbol{y}$，这里 $\boldsymbol{A}$ 为 $m\times m$ 的雅可比矩阵 $\left(\frac{\partial f_i}{\partial y_j}\right)$. 若 $\boldsymbol{A}$ 有 m 个特征值 $\lambda_1,\cdots,\lambda_m$，其中 λ_i 可能是复数，所以，为了使模型方程结果能推广到方程组，方程(4.8)中 λ 为复数. 为保证微分方程本身的稳定性，还应假定 $\mathrm{Re}(\lambda)<0$.

下面先研究欧拉方法的稳定性. 模型方程 $y'=\lambda y$ 的欧拉公式为

$$y_{n+1} = (1+h\lambda)y_n. \tag{4.9}$$

设在节点值 y_n 上有一扰动值 ε_n，它的传播使节点值 y_{n+1} 产生大小为 ε_{n+1} 的扰动值，假设用 $y_n^*=y_n+\varepsilon_n$ 按欧拉公式得出 $y_{n+1}^*=y_{n+1}+\varepsilon_{n+1}$ 的计算过程不再有新的误差，则扰动值满足

$$\varepsilon_{n+1} = (1+h\lambda)\varepsilon_n.$$

可见扰动值满足原来的差分方程(4.9). 这样，如果差分方程的解是不增长的，即有

$$|y_{n+1}| \leqslant |y_n|,$$

则它就是稳定的. 这一论断对于下面将要研究的其他方法同样适用.

显然，为要保证差分方程(4.9)的解是不增长的，只要选取 h 充分小，使

$$|1+h\lambda| \leqslant 1. \tag{4.10}$$

在 $\mu=h\lambda$ 的复平面上，这是以$(-1,0)$为圆心，1 为半径的单位圆域. 称为欧拉法的绝对稳定域，一般情形可由下面定义.

定义 6 单步法(4.1)用于解模型方程(4.8)，若得到的解

$y_{n+1}=E(h\lambda)y_n$，满足$|E(h\lambda)|<1$，则称方法(4.1)是**绝对稳定的**. 在$\mu=h\lambda$的平面上，使$|E(h\lambda)|<1$的变量围成的区域，称为**绝对稳定域**，它与实轴的交称为**绝对稳定区间**.

对欧拉法$E(h\lambda)=1+h\lambda$，其绝对稳定域已由(4.10)给出，绝对稳定区间为$-2<h\lambda<0$，在例5中$\lambda=-100$，$-2<-100h<0$，即$0<h<2/100=0.02$为绝对稳定区间，例4中取$h=0.025$故它是不稳定的，当取$h=0.005$时它是稳定的.

对二阶R-K方法，解模型方程(4.1)可得到

$$y_{n+1}=\left[1+h\lambda+\frac{(h\lambda)^2}{2}\right]y_n,$$

故

$$E(h\lambda)=1+h\lambda+\frac{(h\lambda)^2}{2}.$$

绝对稳定域由$\left|1+h\lambda+\frac{(h\lambda)^2}{2}\right|<1$得到，于是可得绝对稳定区间为$-2<h\lambda<0$，即$0<h<-2/\lambda$. 类似可得三阶及四阶的R-K方法的$E(h\lambda)$分别为

$$E(h\lambda)=1+h\lambda+\frac{(h\lambda)^2}{2!}+\frac{(h\lambda)^3}{3!},$$

$$E(h\lambda)=1+h\lambda+\frac{(h\lambda)^2}{2!}+\frac{(h\lambda)^3}{3!}+\frac{(h\lambda)^4}{4!}.$$

由$|E(h\lambda)|<1$可得到相应的绝对稳定域. 当λ为实数时则得绝对稳定区间. 它们分别为

三阶显式R-K方法：$-2.51<h\lambda<0$，即$0<h<-2.51/\lambda$.

四阶显式R-K方法：$-2.78<h\lambda<0$，即$0<h<-2.78/\lambda$.

从以上讨论可知显式的R-K方法的绝对稳定域均为有限域，都对步长h有限制. 如果h不在所给的绝对稳定区间内，方法就不稳定.

例5　$y'=-20y\quad(0\leqslant x\leqslant 1)$，$y(0)=1$，分别取$h=0.1$及$h=0.2$用经典的四阶R-K方法(3.13)计算.

解 本例 $\lambda=-20$，$h\lambda$ 分别为 -2 及 -4，前者在绝对稳定区间内，后者则不在，用四阶 R-K 方法计算其误差见下表：

x_n	0.2	0.4	0.6	0.8	1.0
$h=0.1$	0.93×10^{-1}	0.12×10^{-1}	0.14×10^{-2}	0.15×10^{-3}	0.17×10^{-4}
$h=0.2$	4.98	25.0	125.0	625.0	3125.0

从以上结果看到，如果步长 h 不满足绝对稳定条件，误差增长很快.

对隐式单步法，可以同样讨论方法的绝对稳定性，例如对后退欧拉法，用它解模型方程可得

$$y_{n+1}=\frac{1}{1-h\lambda}y_n,$$

故

$$E(h\lambda)=\frac{1}{1-h\lambda}.$$

由 $|E(h\lambda)|=\left|\frac{1}{1-h\lambda}\right|<1$ 可得绝对稳定域为 $|1-h\lambda|>1$，它是以(1,0)为圆心，1 为半径的单位圆外部，故绝对稳定区间为 $-\infty<h\lambda<0$. 当 $\lambda<0$ 时，则 $0<h<\infty$，即对任何步长均为稳定的.

对隐式梯形法，它用于解模型方程(4.8)得

$$y_{n+1}=\frac{1+\dfrac{h\lambda}{2}}{1-\dfrac{h\lambda}{2}}y_n,$$

故

$$E(h\lambda)=\frac{1+h\lambda/2}{1-h\lambda/2}.$$

对 $\mathrm{Re}(\lambda)<0$ 有 $|E(h\lambda)|=\left|\frac{1+\dfrac{h\lambda}{2}}{1-\dfrac{h\lambda}{2}}\right|<1$，故绝对稳定域为 $\mu=h\lambda$

的左半平面，绝对稳定区间为$-\infty<h\lambda<0$，即$0<h<\infty$时梯形法均是稳定的.

9.5 线性多步法

在逐步推进的求解过程中，计算y_{n+1}之前事实上已经求出了一系列的近似值$y_0,y_1,\cdots,y_n$，如果充分利用前面多步的信息来预测y_{n+1}，则可以期望会获得较高的精度. 这就是构造所谓线性多步法的基本思想.

构造多步法的主要途径是基于数值积分方法和基于泰勒展开方法，前者可直接由方程(1.1)两端积分后利用插值求积公式得到. 本节主要介绍基于泰勒展开的构造方法.

9.5.1 线性多步法的一般公式

如果计算y_{n+k}时，除用y_{n+k-1}的值，还用到$y_{n+i}(i=0,1,\cdots,k-2)$的值，则称此方法为**线性多步法**. 一般的线性多步法公式可表示为

$$y_{n+k}=\sum_{i=0}^{k-1}\alpha_i y_{n+i}+h\sum_{i=0}^{k}\beta_i f_{n+i},\tag{5.1}$$

其中y_{n+i}为$y(x_{n+i})$的近似，$f_{n+i}=f(x_{n+i},y_{n+i})$，$x_{n+i}=x_0+ih$，$\alpha_i,\beta_i$为常数，$\alpha_0$及$\beta_0$不全为零，则称(5.1)为线性$k$步法，计算时需先给出前面$k$个近似值$y_0,y_1,\cdots,y_{k-1}$，再由(5.1)逐次求出$y_k,y_{k+1},\cdots$. 如果$\beta_k=0$，称(5.1)为显式$k$步法，这时$y_{n+k}$可直接由(5.1)算出；如果$\beta_k\neq0$，则(5.1)称为隐式$k$步法，求解时与梯形法(2.7)相同，要用迭代法方可算出y_{n+k}. (5.1)中系数α_i及β_i可根据方法的局部截断误差及阶确定，其定义为：

定义 7 设$y(x)$是初值问题(1.1)，(1.2)的准确解，线性多步法(5.1)在x_{n+k}上的局部截断误差为

$$T_{n+k}=L[y(x_n);h]$$
$$=y(x_{n+k})-\sum_{i=0}^{k-1}\alpha_i y(x_{n+i})-h\sum_{i=0}^{k}\beta_i y'(x_{n+i}). \quad (5.2)$$

若 $T_{n+k}=O(h^{p+1})$，则称方法(5.1)是 p 阶的，$p\geqslant 1$ 则称方法(5.1)与方程(1.1)是相容的.

由定义 7，对 T_{n+k} 在 x_n 处做泰勒展开，由于

$$y(x_n+ih)=y(x_n)+ihy'(x_n)+\frac{(ih)^2}{2!}y''(x_n)$$
$$+\frac{(ih)^3}{3!}y'''(x_n)+\cdots,$$
$$y'(x_n+ih)=y'(x_n)+ihy''(x_n)+\frac{(ih)^2}{2!}y'''(x_n)+\cdots.$$

代入(5.2)得

$$T_{n+k}=c_0y(x_n)+c_1hy'(x_n)+c_2h^2y''(x_n)$$
$$+\cdots+c_ph^py^{(p)}(x_n)+\cdots, \quad (5.3)$$

其中

$$c_0=1-(\alpha_0+\cdots+\alpha_{k-1}),$$
$$c_1=k-[\alpha_1+2\alpha_2+\cdots+(k-1)\alpha_{k-1}]-(\beta_0+\beta_1+\cdots+\beta_k),$$
$$c_q=\frac{1}{q!}[k^q-(\alpha_1+2^q\alpha_2+\cdots+(k-1)^q\alpha_{k-1}]$$
$$-\frac{1}{(q-1)!}[\beta_1+2^{q-1}\beta_2+\cdots+k^{q-1}\beta_k]$$
$$q=2,3,\cdots. \quad (5.4)$$

若在公式(5.1)中选择系数 α_i 及 β_i，使它满足

$$c_0=c_1=\cdots=c_p=0,\quad c_{p+1}\neq 0.$$

由定义可知此时所构造的多步法是 p 阶的，且

$$T_{n+k}=c_{p+1}h^{p+1}y^{(p+1)}(x_n)+O(h^{p+2}). \quad (5.5)$$

称右端第一项为**局部截断误差主项**，c_{p+1} 称为**误差常数**.

根据相容性定义，$p\geqslant 1$，即 $c_0=c_1=0$，由(5.4)得

$$\begin{cases} \alpha_0 + \alpha_1 + \cdots + \alpha_{k-1} = 1, \\ \sum_{i=1}^{k-1} i\alpha_i + \sum_{i=0}^{k} \beta_i = k. \end{cases} \tag{5.6}$$

故方法(5.1)与微分方程(1.1)相容的充分必要条件是(5.6)成立.

显然,当 $k=1$ 时,若 $\beta_1=0$,则由(5.6)可求得

$$\alpha_0 = 1,\ \beta_0 = 1.$$

此时公式(5.1)为

$$y_{n+1} = y_n + hf_n,$$

即为欧拉法.从(5.4)可求得 $c_2=1/2\neq 0$,故方法为 1 阶精度,且局部截断误差为

$$T_{n+1} = \frac{1}{2}h^2 y''(x_n) + O(h^3),$$

这和第 2 节给出的定义及结果是一致的.

对 $k=1$,若 $\beta_1\neq 0$,此时方法为隐式公式,为了确定系数 α_0,β_0,β_1,可由 $c_0=c_1=c_2=0$ 解得 $\alpha_0=1$,$\beta_0=\beta_1=1/2$.于是得到公式

$$y_{n+1} = y_n + \frac{h}{2}(f_n + f_{n+1}),$$

即为梯形法.由(5.4)可求得 $c_3=-1/12$,故 $p=2$,所以梯形法是二阶方法,其局部截断误差主项是 $-h^3 y'''(x_n)/12$,这与第 2 节中的讨论也是一致的.

对 $k\geqslant 2$ 的多步法公式都可利用(5.4)确定系数 α_i,β_i,并由(5.5)给出局部截断误差,下面只就若干常用的多步法导出具体公式.

9.5.2　阿当姆斯显式与隐式公式

考虑形如

$$y_{n+k} = y_{n+k-1} + h\sum_{i=0}^{k}\beta_i f_{n+i} \tag{5.7}$$

的 k 步法,称为**阿当姆斯(Adams)方法**.$\beta_k=0$ 为显式方法,$\beta_k\neq 0$

为隐式方法，通常称为阿当姆斯显式与隐式公式，也称 Adams-Bashforth 公式与 Adams-Monlton 公式. 这类公式可直接由方程(1.1)两端积分(从 x_{n+k-1} 到 x_{n-k} 积分)求得. 下面可利用(5.4)由 $c_1=\cdots=c_p=0$ 推出，对比(5.7)与(5.1)可知此时系数 $\alpha_0=\alpha_1=\cdots=\alpha_{k-2}=0$，$\alpha_{k-1}=1$，显然 $c_0=0$ 成立，下面只需确定系数 β_0，$\beta_1,\cdots,\beta_k$，故可令 $c_1=\cdots=c_{k+1}=0$，则可求得 $\beta_0,\beta_1,\cdots,\beta_k$(若 $\beta_k=0$，则令 $c_0=\cdots=c_k=0$ 来求得 $\beta_0,\beta_1,\cdots,\beta_{k-1}$). 下面以 $k=3$ 为例，由 $c_1=c_2=c_3=c_4=0$，根据(5.4)得

$$\begin{cases}\beta_0+\beta_1+\beta_2+\beta_3=1,\\ 2(\beta_1+2\beta_2+3\beta_3)=5,\\ 3(\beta_1+4\beta_2+9\beta_3)=19,\\ 4(\beta_1+8\beta_2+27\beta_3)=65.\end{cases}$$

若 $\beta_3=0$，则由前三个方程解得

$$\beta_0=\frac{5}{12},\quad \beta_1=-\frac{16}{12},\quad \beta_2=\frac{23}{12},$$

得到 $k=3$ 的阿当姆斯显式公式是

$$y_{n+3}=y_{n+2}+\frac{h}{12}(23f_{n+2}-16f_{n+1}+5f_n). \tag{5.8}$$

由(5.4)求得 $c_4=3/8$，所以(5.8)是三阶方法，局部截断误差是

$$T_{n+3}=\frac{3}{8}h^4y^{(4)}(x_n)+O(h^5).$$

若 $\beta_3\neq 0$，则可解得

$$\beta_0=\frac{1}{24},\quad \beta_1=-\frac{5}{24},\quad \beta_2=\frac{19}{24},\quad \beta_3=\frac{3}{8}.$$

于是得 $k=3$ 的阿当姆斯隐式公式为

$$y_{n+3}=y_{n+2}+\frac{h}{24}(9f_{n-3}+19f_{n+2}-5f_{n+1}+f_n), \tag{5.9}$$

它是四阶方法，局部截断误差是

$$T_{n+3}=-\frac{19}{720}h^5y^{(5)}(x_n)+O(h^6). \tag{5.10}$$

用类似的方法可求得阿当姆斯显式方法和隐式方法的公式，表 9-5 及表 9-6 分别列出了 $k=1,2,3,4$ 时的阿当姆斯显式公式与阿当姆斯隐式公式，其中 k 为步数，p 为方法的阶，c_{p+1} 为误差常数.

表 9-5 阿当姆斯显式公式

k	p	公 式	c_{p+1}
1	1	$y_{n+1}=y_n+hf_n$	$\frac{1}{2}$
2	2	$y_{n+2}=y_{n+1}+\frac{h}{2}(3f_{n+1}-f_n)$	$\frac{5}{12}$
3	3	$y_{n+3}=y_{n+2}+\frac{h}{12}(23f_{n+2}-16f_{n+1}+5f_n)$	$\frac{3}{8}$
4	4	$y_{n+4}=y_{n+3}+\frac{h}{24}(55f_{n+3}-59f_{n+2}+37f_{n+1}-9f_n)$	$\frac{251}{720}$

表 9-6 阿当姆斯隐式公式

k	p	公 式	c_{p+1}
1	2	$y_{n+1}=y_n+\frac{h}{2}(f_{n+1}+f_n)$	$-\frac{1}{12}$
2	3	$y_{n+2}=y_{n+1}+\frac{h}{12}(5f_{n+2}+8f_{n+1}-f_n)$	$-\frac{1}{24}$
3	4	$y_{n+3}=y_{n+2}+\frac{h}{24}(9f_{n+3}+19f_{n+2}-5f_{n+1}+f_n)$	$-\frac{19}{720}$
4	5	$y_{n+4}=y_{n+3}+\frac{h}{720}(251f_{n+4}+646f_{n+3}-264f_{n+2}$ $+106f_{n+1}-19f_n)$	$-\frac{3}{160}$

例 6 用四阶阿当姆斯显式和隐式方法解初值问题

$$y'=-y+x+1,\quad y(0)=1.$$

取步长 $h=0.1$.

解 本题 $f_n=-y_n+x_n+1$，$x_n=nh=0.1n$. 从四阶阿当姆斯

显式公式得到

$$y_{n+4}=y_{n+3}+\frac{h}{24}(55f_{n+3}-59f_{n+2}+37f_{n+1}-9f_n)$$

$$=\frac{1}{24}(18.5y_{n+3}+5.9y_{n+2}-3.7y_{n+1}+0.9y_n$$

$$+0.24n+3.24).$$

对于四阶阿当姆斯隐式公式得到

$$y_{n+3}=y_{n+2}+\frac{h}{24}(9f_{n+3}+19f_{n+2}-5f_{n+1}+f_n)$$

$$=\frac{1}{24}(-0.9y_{n+3}+22.1y_{n+2}+0.5y_{n+1}-0.1y_n+0.24n+3).$$

由此可直接解出 y_{n+3} 而不用迭代，得到

$$y_{n+3}=\frac{1}{24.9}(22.1y_{n+2}+0.5y_{n+1}-0.1y_n+0.24n+3).$$

计算结果见表 9-7，其中显式方法中的 y_0, y_1, y_2, y_3 及隐式方法中的 y_0, y_1, y_2 均用准确解 $y(x)=e^{-x}+x$ 计算得到，对一般方程，可用四阶 R-K 方法计算初始近似.

表 9-7 计算结果

x_n	精确解 $y(x_n)=e^{-x_n}+x_n$	阿当姆斯显式方法		阿当姆斯隐式方法	
		y_n	$\|y(x_n)-y_n\|$	y_n	$\|y(x_n)-y_n\|$
0.3	1.040 818 22			1.040 818 01	2.1×10^{-7}
0.4	1.070 320 05	1.070 322 92	2.87×10^{-6}	1.070 319 66	3.9×10^{-7}
0.5	1.106 530 66	1.106 535 48	4.82×10^{-6}	1.106 530 14	5.2×10^{-7}
0.6	1.148 811 64	1.148 818 41	6.77×10^{-6}	1.148 811 01	6.3×10^{-7}
0.7	1.196 585 30	1.196 593 40	8.10×10^{-6}	1.196 584 59	7.1×10^{-7}
0.8	1.249 328 96	1.249 338 16	9.20×10^{-6}	1.249 328 19	7.7×10^{-7}
0.9	1.306 569 66	1.306 579 62	9.96×10^{-6}	1.306 568 84	8.2×10^{-7}
1.0	1.367 879 44	1.367 889 96	1.05×10^{-5}	1.367 878 59	8.5×10^{-7}

从以上例子看到同阶的阿当姆斯方法，隐式方法要比显式方法误差小，这可以从两种方法的局部截断误差主项 $c_{p+1}h^{p+1}y^{(p+1)}(x_n)$ 的系数大小得到解释，这里 c_{p+1} 分别为 251/720 及 −19/720.

9.5.3 米尔尼方法与辛普森方法

考虑与(5.7)不同的另一个 $k=4$ 的显式公式

$$y_{n+4} = y_n + h(\beta_3 f_{n+3} + \beta_2 f_{n+2} + \beta_1 f_{n+1} + \beta_0 f_n),$$

其中 $\beta_0,\beta_1,\beta_2,\beta_3$ 为待定常数，可根据使公式的阶尽可能高这一条件来确定其数值. 由(5.4)可知 $c_0=0$，再令 $c_1=c_2=c_3=c_4=0$ 得到

$$\begin{cases}\beta_0+\beta_1+\beta_2+\beta_3 = 4,\\ 2(\beta_1+2\beta_2+3\beta_3) = 16,\\ 3(\beta_1+4\beta_2+9\beta_3) = 64,\\ 4(\beta_1+8\beta_2+27\beta_3) = 256.\end{cases}$$

解此方程组得

$$\beta_3 = \frac{8}{3},\ \beta_2 = -\frac{4}{3},\ \beta_1 = \frac{8}{3},\ \beta_0 = 0.$$

于是得到四步显式公式

$$y_{n+4} = y_n + \frac{4h}{3}(2f_{n+3} - f_{n+2} + 2f_{n+1}), \tag{5.11}$$

称为**米尔尼(Milne)方法**. 由于 $c_5=14/45$，故方法为 4 阶，其局部截断误差为

$$T_{n+4} = \frac{14}{45}h^5 y^{(5)}(x_n) + O(h^6). \tag{5.12}$$

米尔尼方法也可以通过方程(1.1)两端积分

$$y(x_{n+4}) - y(x_n) = \int_{x_n}^{x_{n+4}} f(x,y(x))\mathrm{d}x$$

得到. 若将方程(1.1)从 x_n 到 x_{n+2} 积分，可得

$$y(x_{n+2}) - y(x_n) = \int_{x_n}^{x_{n+2}} f(x, y(x))\,\mathrm{d}x.$$

右端积分通过辛普森求积公式就有

$$y_{n+2} = y_n + \frac{h}{3}(f_n + 4f_{n+1} + f_{n+2}). \tag{5.13}$$

称为**辛普森方法**.它是隐式二步四阶方法,其局部截断误差为

$$T_{n+2} = -\frac{h^5}{90}y^{(5)}(x_n) + O(h^6). \tag{5.14}$$

9.5.4 汉明方法

辛普森公式是二步方法中阶数最高的,但它的稳定性较差,为了改善稳定性,我们考察另一类三步法公式

$$y_{n+3} = \alpha_0 y_n + \alpha_1 y_{n+1} + \alpha_2 y_{n+2} + h(\beta_1 f_{n+1} + \beta_2 f_{n+2} + \beta_3 f_{n+3}),$$

其中系数 $\alpha_0, \alpha_1, \alpha_2$ 及 $\beta_1, \beta_2, \beta_3$ 为常数,如果希望导出的公式是四阶的,则系数中至少有一个自由参数.若取 $\alpha_1 = 1$,则可得到辛普森公式.若取 $\alpha_1 = 0$,仍利用泰勒展开,由(5.4),令 $c_0 = c_1 = c_2 = c_3 = c_4 = 0$,则可得到

$$\begin{cases} \alpha_0 + \alpha_2 = 1, \\ 2\alpha_2 + \beta_1 + \beta_2 + \beta_3 = 3, \\ 4\alpha_2 + 2(\beta_1 + 2\beta_2 + 3\beta_3) = 9, \\ 8\alpha_2 + 3(\beta_1 + 4\beta_2 + 9\beta_3) = 27, \\ 16\alpha_2 + 4(\beta_1 + 8\beta_2 + 27\beta_3) = 81. \end{cases}$$

解此方程组得

$$\alpha_0 = -\frac{1}{8}, \quad \alpha_2 = \frac{9}{8}, \quad \beta_1 = -\frac{3}{8}, \quad \beta_2 = \frac{6}{8}, \quad \beta_3 = \frac{3}{8}.$$

于是有

$$y_{n+3} = \frac{1}{8}(9y_{n+2} - y_n) + \frac{3h}{8}(f_{n+3} + 2f_{n+2} - f_{n+1}), \tag{5.15}$$

称为**汉明(Hamming)方法**.由于 $c_5 = -1/40$,故方法是四阶的,且

局部截断误差为

$$T_{n+3}=-\frac{h^5}{40}y^{(5)}(x_n)+O(h^6). \tag{5.16}$$

9.5.5 预测-校正方法

对于隐式的线性多步法，计算时要进行迭代，计算量较大. 为了避免进行迭代，通常采用显式公式给出 y_{n+k} 的一个初始近似，记为 $y_{n+k}^{(0)}$，称为**预测**(predictor)，接着计算 f_{n+k} 的值(evaluation)，再用隐式公式计算 y_{n+k}，称为**校正**(corrector). 例如在(2.13)中用欧拉法做预测，再用梯形法校正，得到改进欧拉法，它就是一个二阶预测-校正方法. 一般情况下，预测公式与校正公式都取同阶的显式方法与隐式方法相匹配. 例如用四阶的阿当姆斯显式方法做预测，再用四阶阿当姆斯隐式公式做校正，得到以下格式：

预测 P：$y_{n+4}^p=y_{n+3}+\frac{h}{24}(55f_{n+3}-59f_{n+2}+37f_{n+1}-9f_n)$，

求值 E：$f_{n+4}^p=f(x_{n+4},y_{n+4}^p)$，

校正 C：$y_{n+4}=y_{n+3}+\frac{h}{24}(9f_{n+4}^p+19f_{n+3}-5f_{n+2}+f_{n+1})$，

求值 E：$f_{n+4}=f(x_{n+4},y_{n+4})$.

此公式称为**阿当姆斯四阶预测-校正格式**(PECE).

依据四阶阿当姆斯公式的截断误差，对于 PECE 的预测步 P 有

$$y(x_{n+4})-y_{n+4}^p\approx\frac{251}{720}h^5y^{(5)}(x_n),$$

对校正步 C 有

$$y(x_{n+4})-y_{n+4}\approx-\frac{19}{720}h^5y^{(5)}(x_n).$$

两式相减得

$$h^5y^{(5)}(x_n)\approx-\frac{720}{270}(y_{n+4}^p-y_{n+4}),$$

于是有下列事后误差估计

$$y(x_{n+4}) - y_{n+4}^p \approx -\frac{251}{270}(y_{n+4}^p - y_{n+4}),$$

$$y(x_{n+4}) - y_{n+4} \approx \frac{19}{270}(y_{n+4}^p - y_{n+4}).$$

容易看出

$$\left.\begin{aligned} y_{n+4}^{pm} &= y_{n+4}^p + \frac{251}{270}(y_{n+4} - y_{n+4}^p), \\ \bar{y}_{n+4} &= y_{n+4} - \frac{19}{270}(y_{n+4} - y_{n+4}^p). \end{aligned}\right\} \tag{5.17}$$

比 y_{n+4}^p，y_{n+4}更好. 但在 y_{n+4}^{pm}的表达式中 y_{n+4}是未知的，因此计算时用上一步代替，从而构造一种**修正预测-校正格式**(PMECME)：

$$\text{P}: y_{n+4}^p = y_{n+3} + \frac{h}{24}(55f_{n+3} - 59f_{n+2} + 37f_{n+1} - 9f_n),$$

$$\text{M}: y_{n+4}^{pm} = y_{n+4}^p + \frac{251}{270}(y_{n+3}^c - y_{n+3}^p),$$

$$\text{E}: f_{n+4}^{pm} = f(x_{n+4}, y_{n+4}^{pm}),$$

$$\text{C}: y_{n+4}^c = y_{n+3} + \frac{h}{24}(9f_{n+4}^{pm} + 19f_{n+3} - 5f_{n+2} + f_{n+1}),$$

$$\text{M}: y_{n+4} = y_{n+4}^c - \frac{19}{270}(y_{n+4}^c - y_{n+4}^p),$$

$$\text{E}: f_{n+4} = f(x_{n+4}, y_{n+4}).$$

注意：在 PMECME 格式中已将(5.17)的 y_{n+4}及 $\bar{y}_{n+4}$分别改为 y_{n+4}^c及 y_{n+4}.

利用米尔尼公式(5.11)和汉明公式(5.15)相匹配，并利用截断误差(5.12)，(5.16)改进计算结果，可类似地建立四阶**修正米尔尼-汉明预测-校正格式**(PMECME)：

$$\text{P}: y_{n+4}^p = y_n + \frac{4}{3}h(2f_{n+3} - f_{n+2} + 2f_{n+1}),$$

$$\text{M}: y_{n+4}^{pm} = y_{n+4}^p + \frac{112}{121}(y_{n+3}^c - y_{n+3}^p),$$

E：$f_{n+4}^{pm} = f(x_{n+4}, y_{n+4}^{pm})$，

C：$y_{n+4}^{c} = \frac{1}{8}(9y_{n+3} - y_{n+1}) + \frac{3}{8}h(f_{n+4}^{pm} + 2f_{n+3} - f_{n+2})$，

M：$y_{n+4} = y_{n+4}^{c} - \frac{9}{121}(y_{n+4}^{c} - y_{n+4}^{p})$，

E：$f_{n+4} = f(x_{n+4}, y_{n+4})$.

例 7 将例 6 的初值问题用修正的米尔尼-汉明预测-校正公式计算 y_5 及 y_6，初值 y_0, y_1, y_2, y_3 仍用已算出的精确解，即 $y_0 = 1, y_1 = 1.004\,837\,42, y_2 = 1.018\,730\,75, y_3 = 1.040\,818\,22$，给出计算结果及误差.

解 根据修正的米尔尼-汉明预测-校正公式可得

$$y_5^p = 1.106\,532\,99,$$

$$y_5^{pm} = y_5^p + \frac{112}{121} \times (y_4^c - y_4^p) = 1.106\,530\,364.$$

（注：$y_4^p = 1.070\,322\,60, y_4^c = 1.070\,319\,66$）

$$f(x_5, y_5^{pm}) = -y_5^{pm} + x_5 + 1 = 0.393\,469\,636,$$

$$y_5^c = \frac{1}{8} \times (9y_4 - y_2) + \frac{0.3}{8} \times [f(x_5, y_5^{pm}) + 2f_4 - f_3]$$

$$= 1.106\,530\,419,$$

$$y_5 = y_5^c - \frac{9}{121} \times (y_5^c - y_5^p) = 1.106\,530\,61;$$

$$f_5 = -y_5 + x_5 + 1 = 0.393\,469\,39,$$

$$y_6^p = y_2 + \frac{0.4}{3} \times (2f_5 - f_4 + 2f_3) = 1.148\,813\,73,$$

$$y_6^{pm} = y_6^p + \frac{112}{121} \times (y_5^c - y_5^p) = 1.148\,811\,35,$$

$$f_6^{pm} = -y_6^{pm} + x_6 + 1 = 0.451\,188\,65,$$

$$y_6^c = \frac{1}{8} \times (9y_5 - y_3) + \frac{0.3}{8} \times (f_6^{pm} + 2f_5 - f_4) = 1.148\,811\,44,$$

$$y_6 = y_6^c - \frac{9}{121} \times (y_6^c - y_6^p) = 1.148\ 811\ 61.$$

误差

$$|y(x_5) - y_5| = 4.97 \times 10^{-8}, \ |y(x_6) - y_6| = 2.61 \times 10^{-8}.$$

从结果看，此方法的误差比四阶阿当姆斯隐式法和四阶汉明方法小，这与理论分析一致.

9.5.6 构造多步法公式的注记和例

前面已指出构造多步法公式有基于数值积分和泰勒展开两种途径，只对能将微分方程(1.1)转化为等价的积分方程的情形方可利用数值积分方法建立多步法公式，它是有局限性的. 即前种途径只对部分方法适用. 而用泰勒展开则可构造任意多步法公式，其做法是根据多步法公式的形式，直接在 x_n 处做泰勒展开即可. 不必套用系数公式(5.4)确定多步法(5.1)的系数 α_i 及 β_i ($i=0,1,\cdots,k$)，因为多步法公式不一定如(5.1)的形式. 另外，套用公式容易记错. 具体做法见下面例子.

例 8 解初值问题 $y'=f(x,y)$, $y(x_0)=y_0$. 用显式二步法 $y_{n+1}=\alpha_0 y_n+\alpha_1 y_{n-1}+h(\beta_0 f_n+\beta_1 f_{n-1})$，其中 $f_n=f(x_n,y_n)$, $f_{n-1}=f(x_{n-1},y_{n-1})$. 试确定参数 $\alpha_0,\alpha_1,\beta_0,\beta_1$ 使方法阶数尽可能高，并求局部截断误差.

解 本题仍根据局部截断误差定义，用泰勒展开确定参数满足的方程. 由于

$$\begin{aligned}
T_{n+1} &= y(x_n+h) - \alpha_0 y(x_n) - \alpha_1 y(x_n - h) \\
&\quad - h[\beta_0 y'(x_n) + \beta_1 y'(x_n - h)] \\
&= y(x_n) + hy'(x_n) + \frac{h^2}{2}y''(x_n) + \frac{h^3}{3!}y'''(x_n) \\
&\quad + \frac{h^4}{4!}y^{(4)}(x_n) + O(h^5) - \alpha_0 y(x_n) \\
&\quad - \alpha_1\Big[y(x_n) - hy'(x_n) + \frac{h^2}{2}y''(x_n)
\end{aligned}$$

$$- \frac{h^3}{3!}y'''(x_n) + \frac{h^4}{4!}y^{(4)}(x_n) + O(h^5)\Big]$$

$$-\beta_0 h y'(x_n) - \beta_1 h\Big[y'(x_n) - hy''(x_n)$$

$$+ \frac{h^2}{2}y'''(x_n) - \frac{h^3}{3!}y^{(4)}(x_n) + O(h^4)\Big]$$

$$= (1-\alpha_0-\alpha_1)y(x_n) + (1+\alpha_1-\beta_0-\beta_1)hy'(x_n)$$

$$+ \left(\frac{1}{2} - \frac{1}{2}\alpha_1 + \beta_1\right)h^2 y''(x_n) + \left(\frac{1}{6} + \frac{1}{6}\alpha_1 - \frac{1}{2}\beta_1\right)h^3$$

$$\cdot y'''(x_n) + \left(\frac{1}{24} - \frac{1}{24}\alpha_1 + \frac{1}{6}\beta_1\right)h^4 y^{(4)}(x_n) + O(h^5)$$

为求参数 $\alpha_0,\alpha_1,\beta_0,\beta_1$ 使方法阶数尽量高，可令

$$1-\alpha_0-\alpha_1 = 0,\quad 1+\alpha_1-\beta_0-\beta_1 = 0,$$

$$\frac{1}{2} - \frac{1}{2}\alpha_1 + \beta_1 = 0,\quad \frac{1}{6} + \frac{1}{6}\alpha_1 - \frac{1}{2}\beta_1 = 0.$$

即得方程组

$$\begin{cases} \alpha_0 + \alpha_1 = 1, \\ -\alpha_1 + \beta_0 + \beta_1 = 1, \\ \alpha_1 - 2\beta_1 = 1, \\ -\alpha_1 + 3\beta_1 = 1. \end{cases}$$

解得 $\alpha_0 = -4, \alpha_1 = 5, \beta_0 = 4, \beta_1 = 2$，此时公式为三阶，而且

$$T_{n+1} = \frac{1}{6}h^4 y^{(4)}(x_n) + O(h^5)$$

即为所求局部截断误差. 而所得二步法为

$$y_{n+1} = -4y_n + 5y_{n-1} + 2h(2f_n + f_{n-1}).$$

例 9　证明存在 α 的一个值，使线性多步法

$$y_{n+1} + \alpha(y_n - y_{n-1}) - y_{n-2} = \frac{1}{2}(3+\alpha)h(f_n + f_{n-1})$$

是四阶的.

证明　只要证明局部截断误差 $T_{n+1} = O(h^5)$，则方法为四阶. 仍用泰勒展开，由于

$$
\begin{aligned}
T_{n+1} = & y(x_n+h)+\alpha[y(x_n)-y(x_n-h)]-y(x_n-2h) \\
& -\frac{1}{2}(3+\alpha)h[y'(x_n)+y'(x_n-h)] \\
= & y(x_n)+hy'(x_n)+\frac{h^2}{2}y''(x_n)+\frac{h^3}{3!}y'''(x_n)+\frac{h^4}{4!}y^{(4)}(x_n) \\
& +O(h^5)-\alpha\left[(-h)y'(x_n)+\frac{h^2}{2}y''(x_n)-\frac{h^3}{3!}y'''(x_n)\right. \\
& \left.+\frac{h^4}{4!}y^{(4)}(x_n)+O(h^5)\right]-\left[y(x_n)-2hy'(x_n)\right. \\
& \left.+\frac{(2h)^2}{2}y''(x_n)-\frac{(2h)^3}{3!}y'''(x_n)+\frac{(2h)^4}{4!}y^{(4)}(x_n)+O(h^5)\right] \\
& -\frac{h}{2}(3+\alpha)\left[y'(x_n)+y'(x_n)-hy''(x_n)+\frac{h^2}{2}y'''(x_n)\right. \\
& \left.-\frac{h^3}{3!}y^{(4)}(x_n)+O(h^4)\right] \\
= & [1+\alpha+2-(3+\alpha)]hy'(x_n)+\left[\frac{1}{2}-\frac{1}{2}\alpha-2\right. \\
& \left.+\frac{1}{2}(3+\alpha)\right]h^2y''(x_n)+\left[\frac{1}{6}+\frac{1}{6}\alpha+\frac{4}{3}\right. \\
& \left.-\frac{1}{4}(3+\alpha)\right]h^3y'''(x_n)+\left[\frac{1}{24}-\frac{1}{24}\alpha-\frac{2}{3}+\frac{1}{12}(3+\alpha)\right] \\
& \times h^4y^{(4)}(x_n)+O(h^5) \\
= & \left(\frac{3}{4}-\frac{1}{12}\alpha\right)h^3y'''(x_n)+\frac{1}{24}(-9+\alpha)h^4y^{(4)}(x_n)+O(h^5).
\end{aligned}
$$

当 $\alpha=9$ 时，$T_{n+1}=O(h^5)$，故方法是四阶的.

9.6 方程组和高阶方程

9.6.1 一阶方程组

前面我们研究了单个方程 $y'=f$ 的数值解法，只要把 y 和 f

理解为向量,那么,所提供的各种计算公式即可应用到一阶方程组的情形.

考察一阶方程组

$$y'_i = f_i(x, y_1, y_2, \cdots, y_N) \quad (i = 1, 2, \cdots, N)$$

的初值问题,初始条件给为

$$y_i(x_0) = y_i^0 \qquad (i = 1, 2, \cdots, N).$$

若采用向量的记号,记

$$\boldsymbol{y} = (y_1, y_2, \cdots, y_N)^{\mathrm{T}},$$
$$\boldsymbol{y}^0 = (y_1^0, y_2^0, \cdots, y_N^0)^{\mathrm{T}},$$
$$\boldsymbol{f} = (f_1, f_2, \cdots, f_N)^{\mathrm{T}}.$$

则上述方程组的初值问题可表示为

$$\left.\begin{aligned} \boldsymbol{y}' &= \boldsymbol{f}(x, \boldsymbol{y}), \\ \boldsymbol{y}(x_0) &= \boldsymbol{y}_0. \end{aligned}\right\} \tag{6.1}$$

求解这一初值问题的四阶龙格-库塔公式为

$$\boldsymbol{y}_{n+1} = \boldsymbol{y}_n + \frac{h}{6}(\boldsymbol{k}_1 + 2\boldsymbol{k}_2 + 2\boldsymbol{k}_3 + \boldsymbol{k}_4),$$

式中

$$\boldsymbol{k}_1 = \boldsymbol{f}(x_n, \boldsymbol{y}_n),$$
$$\boldsymbol{k}_2 = \boldsymbol{f}\left(x_n + \frac{h}{2}, \boldsymbol{y}_n + \frac{h}{2}\boldsymbol{k}_1\right),$$
$$\boldsymbol{k}_3 = \boldsymbol{f}\left(x_n + \frac{h}{2}, \boldsymbol{y}_n + \frac{h}{2}\boldsymbol{k}_2\right),$$
$$\boldsymbol{k}_4 = \boldsymbol{f}(x_n + h, \boldsymbol{y}_n + h\boldsymbol{k}_3).$$

或表示为

$$y_{i,n+1} = y_{in} + \frac{h}{6}(K_{i1} + 2K_{i2} + 2K_{i3} + K_{i4})$$
$$(i = 1, 2, \cdots, N),$$

其中

$$K_{i1} = f_i(x_n, y_{1n}, y_{2n}, \cdots, y_{Nn}),$$

$$K_{i2}=f_i\left(x_n+\frac{h}{2},y_{1n}+\frac{h}{2}K_{11},y_{2n}+\frac{h}{2}K_{21},\cdots,y_{Nn}+\frac{h}{2}K_{N1}\right),$$

$$K_{i3}=f_i\left(x_n+\frac{h}{2},y_{1n}+\frac{h}{2}K_{12},y_{2n}+\frac{h}{2}K_{22},\cdots,y_{Nn}+\frac{h}{2}K_{N2}\right),$$

$$K_{i4}=f_i(x_n+h,y_{1n}+hK_{13},y_{2n}+hK_{23},\cdots,y_{Nn}+hK_{N3}).$$

这里 y_{in} 是第 i 个因变量 $y_i(x)$ 在节点 $x_n=x_0+nh$ 的近似值.

为了帮助理解这一公式的计算过程,我们再考察两个方程的特殊情形:

$$\begin{cases} y'=f(x,y,z), \\ z'=g(x,y,z), \\ y(x_0)=y_0, \\ z(x_0)=z_0. \end{cases}$$

这时四阶龙格-库塔公式具有形式

$$\left.\begin{aligned} y_{n+1}&=y_n+\frac{h}{6}(K_1+2K_2+2K_3+K_4), \\ z_{n+1}&=z_n+\frac{h}{6}(L_1+2L_2+2L_3+L_4). \end{aligned}\right\} \tag{6.2}$$

其中

$$\left.\begin{aligned} K_1&=f(x_n,y_n,z_n), \\ K_2&=f\left(x_n+\frac{h}{2},y_n+\frac{h}{2}K_1,z_n+\frac{h}{2}L_1\right), \\ K_3&=f\left(x_n+\frac{h}{2},y_n+\frac{h}{2}K_2,z_n+\frac{h}{2}L_2\right), \\ K_4&=f(x_n+h,y_n+hK_3,z_n+hL_3), \\ L_1&=g(x_n,y_n,z_n), \\ L_2&=g\left(x_n+\frac{h}{2},y_n+\frac{h}{2}K_1,z_n+\frac{h}{2}L_1\right), \\ L_3&=g\left(x_n+\frac{h}{2},y_n+\frac{h}{2}K_2,z_n+\frac{h}{2}L_2\right), \\ L_4&=g(x_n+h,y_n+hK_3,z_n+hL_3). \end{aligned}\right\} \tag{6.3}$$

这是一步法，利用节点 x_n 上的值 y_n, z_n，由(6.3)式顺序计算 $K_1, L_1, K_2, L_2, K_3, L_3, K_4, L_4$，然后代入(6.2)式即可求得节点 x_{n+1} 上的 y_{n+1}, z_{n+1}.

9.6.2 化高阶方程为一阶方程组

关于高阶微分方程(或方程组)的初值问题，原则上总可以归结为一阶方程组来求解. 例如，考察下列 m 阶微分方程

$$y^{(m)} = f(x, y, y', \cdots, y^{(m-1)}), \tag{6.4}$$

初始条件为

$$y(x_0) = y_0, y'(x_0) = y'_0, \cdots, y^{(m-1)}(x_0) = y_0^{(m-1)}. \tag{6.5}$$

只要引进新的变量

$$y_1 = y, y_2 = y', \cdots, y_m = y^{(m-1)},$$

即可将 m 阶方程(6.4)化为如下的一阶方程组：

$$\left.\begin{aligned} &y'_1 = y_2, \\ &y'_2 = y_3, \\ &\cdots\cdots\cdots \\ &y'_{m-1} = y_m, \\ &y'_m = f(x, y_1, y_2, \cdots, y_m). \end{aligned}\right\} \tag{6.6}$$

初始条件(6.5)则相应地化为

$$y_1(x_0) = y_0, y_2(x_0) = y'_0, \cdots, y_m(x_0) = y_0^{(m-1)}. \tag{6.7}$$

不难证明初值问题(6.4)，(6.5)和(6.6)，(6.7)是彼此等价的.

特别地，对于下列二阶方程的初值问题：

$$\begin{cases} y'' = f(x, y, y'), \\ y(x_0) = y_0, \\ y'(x_0) = y'_0. \end{cases}$$

引进新的变量 $z = y'$，即可化为下列一阶方程组的初值问题：

$$\begin{cases} y' = z, \\ z' = f(x,y,z), \\ y(x_0) = y_0, \\ z(x_0) = y'_0. \end{cases}$$

针对这个问题应用四阶龙格-库塔公式(6.2),有

$$\begin{cases} y_{n+1} = y_n + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4), \\ z_{n+1} = z_n + \frac{h}{6}(L_1 + 2L_2 + 2L_3 + L_4). \end{cases}$$

由(6.3)式可得

$$K_1 = z_n, \qquad L_1 = f(x_n, y_n, z_n);$$

$$K_2 = z_n + \frac{h}{2}L_1, L_2 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_1, z_n + \frac{h}{2}L_1\right);$$

$$K_3 = z_n + \frac{h}{2}L_2, L_3 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_2, z_n + \frac{h}{2}L_2\right);$$

$$K_4 = z_n + hL_3, L_4 = f(x_n + h, y_n + hK_3, z_n + hL_3).$$

如果消去 K_1, K_2, K_3, K_4,则上述格式可表示为

$$\begin{cases} y_{n+1} = y_n + hz_n + \frac{h^2}{6}(L_1 + L_2 + L_3), \\ z_{n+1} = z_n + \frac{h}{6}(L_1 + 2L_2 + 2L_3 + L_4). \end{cases}$$

这里

$$L_1 = f(x_n, y_n, z_n),$$

$$L_2 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}z_n, z_n + \frac{h}{2}L_1\right),$$

$$L_3 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}z_n + \frac{h^2}{4}L_1, z_n + \frac{h}{2}L_2\right),$$

$$L_4 = f\left(x_n + h, y_n + hz_n + \frac{h^2}{2}L_2, z_n + hL_3\right).$$

9.6.3 刚性方程组

在求解方程组(6.1)时,经常出现解的分量数量级差别很大的情形,这给数值求解带来很大困难,这种问题称为**刚性**(stiff)**问题**,在化学反应、电子网络和自动控制等领域中都是常见的,先考察以下例子.

给定系统

$$\left.\begin{aligned} u' &= -1000.25u + 999.75v + 0.5, \\ v' &= 999.75u - 1000.25v + 0.5, \\ u(0) &= 1, \\ v(0) &= -1. \end{aligned}\right\} \tag{6.8}$$

它可用解析方法求出准确解,方程右端系数矩阵

$$\boldsymbol{A} = \begin{pmatrix} -1000.25 & 999.75 \\ 999.75 & -1000.25 \end{pmatrix}$$

的特征值为 $\lambda_1 = -0.5$, $\lambda_2 = -2000$,方程的准确解为

$$\begin{cases} u(t) = -\mathrm{e}^{-0.5t} + \mathrm{e}^{-2000t} + 1, \\ v(t) = -\mathrm{e}^{-0.5t} - \mathrm{e}^{-2000t} + 1. \end{cases}$$

当 $t \to \infty$ 时,$u(t) \to 1$,$v(t) \to 1$ 称为稳态解,u,v 中均含有快变分量 e^{-2000t} 及慢变分量 $\mathrm{e}^{-0.5t}$. 对应于 λ_2 的快速衰减的分量在 $t=0.005$ 秒时已衰减到 $\mathrm{e}^{-10} \approx 0$,称 $\tau_2 = -\frac{1}{\lambda_2} = \frac{1}{2000} = 0.0005$ 为**时间常数**. 当 $t = 10\tau_2$ 时快变分量即可被忽略,而对应于 λ_1 的慢变分量,它的时间常数 $\tau_1 = -\frac{1}{\lambda_1} = \frac{1}{0.5} = 2$,它要计算到 $t = 10\tau_1 = 20$ 时,才能衰减到 $\mathrm{e}^{-10} \approx 0$,也就是说解 u,v 必须计算到 $t=20$ 才能达到稳态解. 它表明方程(6.8)的解分量变化速度相差很大,是一个刚性方程组. 如果用四阶龙格-库塔法求解. 步长选取要满足 $h < -2.78/\lambda$,即 $h < -2.78/\lambda_2 = 0.00139$,才能使计算稳定. 而要计算到稳态解至少需要算到 $t=20$,则需计算 14 388 步. 这种用小步长计算长

区间的现象是刚性方程数值求解出现的困难，它是系统本身病态性质引起的.

对一般的线性系统

$$\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t} = \boldsymbol{A}\boldsymbol{y}(t) + \boldsymbol{g}(t), \tag{6.9}$$

其中 $\boldsymbol{y}=(y_1,\cdots,y_N)^{\mathrm{T}}\in\mathbf{R}^N$，$\boldsymbol{g}=(g_1,\cdots,g_N)^{\mathrm{T}}\in\mathbf{R}^N$，$\boldsymbol{A}\in\mathbf{R}^{N\times N}$. 若 $\boldsymbol{A}$ 的特征值 $\lambda_j=\alpha_j+\mathrm{i}\beta_j$ $(j=1,\cdots,N,\mathrm{i}=\sqrt{-1})$ 相应的特征向量为 $\boldsymbol{\varphi}_j$ $(j=1,\cdots,N)$，则方程组(6.9)的通解为

$$\boldsymbol{y}(t) = \sum_{j=1}^{N} c_j \mathrm{e}^{\lambda_j t}\boldsymbol{\varphi}_j + \boldsymbol{\psi}(t), \tag{6.10}$$

其中 c_j 为任意常数，可由初始条件 $y(a)=y^0$ 确定，$\boldsymbol{\psi}(t)$ 为特解.

假定 λ_j 的实部 $\alpha_j=\mathrm{Re}(\lambda_j)<0$，则当 $t\to\infty$ 时，$\boldsymbol{y}(t)\to\boldsymbol{\psi}(t)$，$\boldsymbol{\psi}(t)$ 为稳态解.

定义 8 若线性系统(6.9)中 $\boldsymbol{A}$ 的特征值 λ_j 满足条件 $\mathrm{Re}(\lambda_j)<0$ $(j=1,\cdots,N)$，且

$$s = \max_{1\leqslant j\leqslant N} |\mathrm{Re}(\lambda_j)| \Big/ \min_{1\leqslant j\leqslant N} |\mathrm{Re}(\lambda_j)| \gg 1,$$

则称系统(6.9)为**刚性方程**，称 s 为**刚性比**.

刚性比 $s\gg1$ 时，$\boldsymbol{A}$ 为病态矩阵，故刚性方程也称病态方程. 通常 $s\geqslant10$ 就认为是刚性的. s 越大病态越严重. 方程(6.8)的刚性比 $s=4000$，故它是刚性的.

对一般非线性方程组(6.1)，可类似定义 8，将 $\boldsymbol{f}$ 在点 $(t,\boldsymbol{y}(t))$ 处线性展开，记 $\boldsymbol{J}(t)=\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{y}}\in\mathbf{R}^{N\times N}$，假定 $\boldsymbol{J}(t)$ 的特征值为 $\lambda_j(t)$，$j=1,\cdots,N$，于是由定义 8 可知，当 $\lambda_j(t)$ 满足条件 $\mathrm{Re}(\lambda_j(t))<0$ $(j=1,\cdots,N)$，且

$$s(t) = \max_{1\leqslant j\leqslant N} |\mathrm{Re}(\lambda_j(t))| \Big/ \min_{1\leqslant j\leqslant N} |\mathrm{Re}(\lambda_j(t))| \gg 1,$$

则称系统(6.1)是刚性的，$s(t)$ 称为方程(6.1)的局部刚性比.

求刚性方程数值解时，若用步长受限制的方法就将出现小步

长计算大区间的问题，因此最好使用对步长 h 不加限制的方法，如前面已介绍的欧拉后退法及梯形法，即 A-稳定的方法，所谓 A-稳定就是指数值方法的绝对稳定域包含了 $\mu=h\lambda$ 平面的左半平面. 这种方法当然对步长 h 没有限制，但 A-稳定方法要求太苛刻，Dahlquist 已证明所有显式方法都不是 A-稳定的，而隐式的 A-稳定多步法阶数最高为 2，且以梯形法误差常数为最小. 这就表明本章所介绍的方法中能用于解刚性方程的方法很少. 通常求解刚性方程的高阶线性多步法是**吉尔(Gear)方法**，还有隐式龙格-库塔法(见文献[21])，这些方法都有现成的数学软件可供使用. 本书不再介绍.

评 注

本章研究求解常微分方程初值问题的数值方法. 构造数值方法主要有两条途径：基于数值积分的构造方法和基于泰勒展开的构造方法. 后一种方法更灵活，也更具有一般性. 泰勒展开方法还有一个优点，它在构造差分公式的同时可以得到关于截断误差的估计.

基于泰勒展开构造出的四阶龙格-库塔方法(见 3.3 节)则是计算机上的常用算法. 四阶龙格-库塔方法的优点是精度高，程序简单，计算过程稳定，并且易于调节步长.

四阶龙格-库塔方法也有不足之处：它要求函数 $f(x,y)$ 具有较高的光滑性. 如果 $f(x,y)$ 的光滑性差，那么，它的精度可能还不如欧拉公式(2.1 节)或改进的欧拉公式(2.4 节).

四阶龙格-库塔方法的另一个缺点是计算量比较大，需要耗费较多的机器时间(每一步需四次计算函数 $f(x,y)$ 的值). 相比之下，汉明方法(5.4 节)可以节省计算量(每一步只需两次计算函数 $f(x,y)$ 的值). 但汉明方法是一种四步法，它不是自开始的，需要

借助于四阶龙格-库塔方法提供开始值.

对数值方法的分析还涉及到局部截断误差，整体误差，收敛性，相容性及稳定性等概念，特别是有关绝对稳定性的讨论，它涉及计算时步长的选取，如步长选得不合适，舍入误差恶性增长，结果完全错误. 本章只对单步法的收敛性和稳定性做了讨论. 对多步法的有关理论没有涉及，读者可参阅文献[2]及[21].

刚性方程组是具有重要应用价值的问题，它的理论和解法很多，并且还在进一步发展，详细介绍可参阅文献[21]和[22].

习　题

1. 用欧拉法解初值问题

$$y' = x^2 + 100y^2, y(0) = 0.$$

取步长 $h=0.1$，计算到 $x=0.3$(保留到小数点后 4 位).

2. 用改进欧拉法和梯形法解初值问题

$$y' = x^2 + x - y, y(0) = 0.$$

取步长 $h=0.1$，计算到 $x=0.5$，并与准确解 $y=-e^{-x}+x^2-x+1$ 相比较.

3. 用梯形方法解初值问题

$$\begin{cases} y' + y = 0, \\ y(0) = 1. \end{cases}$$

证明其近似解为

$$y_n = \left(\frac{2-h}{2+h}\right)^n,$$

并证明当 $h\to 0$ 时，它收敛于原初值问题的准确解 $y=e^{-x}$.

4. 利用欧拉方法计算积分

$$\int_0^x e^{t^2}\,dt$$

在点 $x=0.5,1,1.5,2$ 的近似值.

5. 取 $h=0.2$，用四阶经典的龙格-库塔方法求解下列初值问题：

1) $\begin{cases} y'=x+y, & 0<x<1; \\ y(0)=1; \end{cases}$

2) $\begin{cases} y'=3y/(1+x), & 0<x<1; \\ y(0)=1. \end{cases}$

6. 证明对任意参数 t，下列龙格-库塔公式是二阶的：

$$\begin{cases} y_{n+1} = y_n + \dfrac{h}{2}(K_2 + K_3), \\ K_1 = f(x_n, y_n), \\ K_2 = f(x_n + th, y_n + thK_1), \\ K_3 = f(x_n + (1-t)h, y_n + (1-t)hK_1). \end{cases}$$

7. 证明中点公式(3.10)是二阶的，并求其绝对稳定区间.

8. 对于初值问题

$$y' = -100(y - x^2) + 2x,\ y(0) = 1.$$

(1) 用欧拉法求解，步长 h 取什么范围的值，才能使计算稳定.

(2) 若用四阶龙格-库塔法计算，步长 h 如何选取？

(3) 若用梯形公式计算，步长 h 有无限制.

9. 分别用二阶显式阿当姆斯方法和二阶隐式阿当姆斯方法解下列初值问题：

$$y' = 1 - y, y(0) = 0.$$

取 $h=0.2, y_0=0, y_1=0.181$，计算 $y(1.0)$ 并与准确解 $y=1-e^{-x}$ 相比较.

10. 证明解 $y'=f(x,y)$ 的下列差分公式

$$y_{n+1} = \frac{1}{2}(y_n + y_{n-1}) + \frac{h}{4}(4y'_{n+1} - y'_n + 3y'_{n-1})$$

是二阶的，并求出截断误差的主项.

11. 试证明线性二步法

$$y_{n+2} + (b-1)y_{n+1} - by_n = \frac{h}{4}[(b+3)f_{n+2} + (3b+1)f_n]$$

当 $b\neq-1$ 时方法为二阶，当 $b=-1$ 时方法为三阶.

12. 求方程

$$\begin{cases} u' = -10u + 9v, \\ v' = 10u - 11v. \end{cases}$$

的刚性比，用四阶 R-K 方法求解时，最大步长能取多少？

计算实习题

要求：

(1) 用 Matlab 语言或你熟悉的其他算法语言编程序，使之尽量具有通用性.

(2) 上机前充分准备，复习有关算法，写出计算步骤，反复查对程序.

(3) 完成计算后写出实验报告，内容包括：计算机型号及所用算法语言，CPU 时间，算法步骤叙述，变量说明，程序清单，输出计算结果，结果分析和小结等.

(4) 根据教师要求选做下列计算实习题中的 3～5 个题目.

1. 用三次样条插值的三弯矩法，编制第一与第二种边界条件的程序. 设已知数据如下：

x_i	0.2	0.4	0.6	0.8	1.0
$f(x_i)$	0.9798652	0.9177710	0.8080348	0.6386093	0.3843735

求 $f(x)$ 的三次样条插值函数 $S(x)$，满足：

(1) 自然边界条件 $S''(0.2)=S''(1.0)=0$；

(2) 第一种边界条件 $S'(0.2)=0.20271$，$S'(1.0)=1.55741$.

要求输出用追赶法解出的弯矩向量 $(M_0,M_1,\cdots,M_4)$ 和 $S(0.2+0.1i)(i=0,1,\cdots,8)$ 的值. 并画出 $y=S(x)$ 的图形.

2. 编制以离散点 $\{x_i\}_{i=0}^{m}$ 的正交多项式 $\{P_k(x)\}$ 为基的最小二乘拟合程序，并用于对下列数据做三次多项式最小二乘拟合.

x_i	-1.0	-0.5	0.0	0.5	1.0	1.5	2.0
y_i	-4.447	-0.452	0.551	0.048	-0.447	0.549	4.552

取权 $\omega(x_i)\equiv 1$，求出拟合曲线 $y=S(x)=\sum_{k=0}^{3}\alpha_k^* P_k(x)=\sum_{k=0}^{3}a_k x^k$，输出 α_k^*，$P_k(x)$，$a_k(k=0,1,2,3)$ 及平方误差 $\|\delta\|_2^2$，并画出 $y=S(x)$ 的图形.

3. 给出积分

(i) $I=\int_0^2 x^2 e^{-x^2}dx$，(ii) $I=\int_{\frac{\pi}{2}}^{\frac{3}{4}\pi}\cot x dx$，

(iii) $I=\int_2^3 \frac{1}{x^2-1}dx$.

实验要求：

(1) 用龙贝格求积计算上述积分 I 的值，要求到 $|T_{k-1}^{(0)}-T_k^{(0)}|<10^{-6}$ 时结束，输出 T 表及 I 的近似值.

(2) 用 5 点高斯求积公式及复化 3 点高斯求积公式计算上述积分，并输出 I 的近似值.

(3) 分析比较各种计算结果.

4. 比较求一阶导数的数值方法，给出函数

$$f(x)=e^{\frac{1}{x}},\ x\in[0.5,2].$$

实验要求：利用某距离点函数值，必要时给定端点导数值，分别用中心差分，数值积分求导，三次样条求导和理查森外推计算 $f(x)$ 的一阶导数，分析，比较各种方法的效果，说明精度与步长 h 的关系.

5. 给定方程组

(i) $\begin{pmatrix}3.01 & 6.03 & 1.99\\ 1.27 & 4.16 & -1.23\\ 0.987 & -4.81 & 9.34\end{pmatrix}\begin{pmatrix}x_1\\ x_2\\ x_3\end{pmatrix}=\begin{pmatrix}1\\ 1\\ 1\end{pmatrix}$；

(ii) $$\begin{pmatrix} 10 & -7 & 0 & 1 \\ -3 & 2.099999 & 6 & 2 \\ 5 & -1 & 5 & -1 \\ 2 & 1 & 0 & 2 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 8 \\ 5.900001 \\ 5 \\ 1 \end{pmatrix}.$$

实验要求:

(1) 用$\boldsymbol{LU}$分解和列主元高斯消去法求解上述两个方程组. 输出$\boldsymbol{Ax}=\boldsymbol{b}$中矩阵$\boldsymbol{A}$及向量$\boldsymbol{b}$,$\boldsymbol{A}=\boldsymbol{LU}$分解的$\boldsymbol{L}$与$\boldsymbol{U}$,$\det\boldsymbol{A}$及解向量$\boldsymbol{x}$.

(2) 将方程组(i)中系数3.01改为3.00,0.987改为0.990. 用列主元高斯消去法求解,输出列主元行交换次序、解向量$\boldsymbol{x}$及$\det\boldsymbol{A}$,并与(1)中结果比较.

(3) 将方程组(ii)中的2.099999改为2.1,5.900001改为5.9.用列主元高斯消去法求解,输出解向量$\boldsymbol{x}$及$\det\boldsymbol{A}$,并与(1)中结果比较.

6. 研究解线性方程组$\boldsymbol{Ax}=\boldsymbol{b}$迭代法收敛速度. 给定$\boldsymbol{A}\in\mathbf{R}^{20\times 20}$为五对角矩阵

$$\boldsymbol{A} = \begin{pmatrix} 3 & -1/2 & -1/4 & & & & \\ -1/2 & 3 & -1/2 & -1/4 & & & \\ -1/4 & -1/2 & 3 & -1/2 & -1/4 & & \\ & \ddots & \ddots & \ddots & \ddots & & \\ & -1/4 & -1/2 & 3 & -1/2 & -1/4 \\ & & -1/4 & -1/2 & 3 & -1/2 \\ & & & -1/4 & -1/2 & 3 \end{pmatrix}.$$

实验要求:

(1) 选取不同的初始向量$\boldsymbol{x}^{(0)}$及右端项向量$\boldsymbol{b}$,给定迭代误差要求,用雅可比迭代和高斯-塞德尔迭代法求解,观察得到的序列是否收敛? 若收敛,记录迭代次数,分析计算结果并得出你的结论.

(2) 用 SOR 迭代法求上述方程组的解，松弛系数 ω 取 $1<\omega<2$ 的不同值，在 $\|\boldsymbol{x}^{(k)}-\boldsymbol{x}^{(k+1)}\|\leqslant 10^{-5}$ 时停止迭代. 记录迭代次数. 分析计算结果并得出你的结论.

7. 求非线性方程及方程组的根，准确到 10^{-6}. 给定方程分别为

(i) $x^2-3x+2-e^x=0$.

(ii) $\begin{cases}3x_1^2-x_2^2=0,\\ 3x_1x_2^2-x_1^3-1=0,\end{cases}\quad \boldsymbol{x}^{(0)}=(1,1)^{\mathrm{T}}$.

实验要求：

(1) 用你自己设计的一种线性收敛迭代法求方程(i)的根，然后再用斯蒂芬森加速迭代计算.

(2) 用牛顿法求方程(i)的根，输出迭代初值，各次迭代值及迭代次数，并与(1)的结果比较.

(3) 用牛顿法求(ii)的解，输出迭代次数及解向量 $\boldsymbol{x}$ 的近似.

8. 用 QR 算法求矩阵特征值：

$$\text{(i)}\ (\boldsymbol{A})=\begin{pmatrix}6&2&1\\2&3&1\\1&1&1\end{pmatrix},\ \text{(ii)}\ \boldsymbol{H}=\begin{pmatrix}2&3&4&5&6\\4&4&5&6&7\\0&3&6&7&8\\0&0&2&8&9\\0&0&0&1&0\end{pmatrix}.$$

实验要求：

(1) 根据 QR 算法原理编制求(i)及(ii)中矩阵全部特征值的程序并输出计算结果(要求误差 $<10^{-5}$).

(2) 直接用现有数学软件求(i),(ii)的全部特征值，并与(1)的结果比较.

9. 求初值问题的数值解，给定初值问题为

(i) $\begin{cases}y'=\dfrac{1}{x^2}-\dfrac{y}{x},\ 1\leqslant x\leqslant 2,\\ y(1)=1.\end{cases}$

(ii) $\begin{cases} y' = -50y + 50x^2 + 2x, \ 0 \leqslant x \leqslant 1, \\ y(0) = \dfrac{1}{3}. \end{cases}$

实验要求：

(1) 用改进欧拉法（取 $h=0.05$）及四阶 R-K. 方法（取 $h=0.1$）求(i)的数值解，并输出 $x_i=1+0.1i(i=0,1,\cdots,10)$ 的数值解 y_i.

(2) 用经典四阶 R-K 方法解(ii)，步长 h 分别取为 $h=0.1$，0.025，0.01 计算，并打印 $x_i=0.1i(i=0,1,\cdots,10)$ 各点的数值解 y_i 及准确解 $y(x_i)$，并分析结果.

$\left(\text{初值问题(ii)的准确解 } y(x)=\dfrac{1}{3}e^{-50x}+x^2\right)$

附录　并行算法及其基本概念

由于并行计算机的逐步普及，很多高校都配备了大型的多处理机，学习并掌握并行算法对进行大规模的科学与工程计算是十分必要和有益的．为使读者对并行算法有初步了解，特在附录中对并行算法的基本概念做简单介绍，需要进一步了解学习的可参看文献[23]及[24]．

1　并行算法及其分类

传统计算机采用冯·诺伊曼(Von Neumann)结构，其特点是每一时刻按一条指令加工处理一个或一对数据，这类计算机称作单指令流单数据流系统，简称 SISD(Single Instruction Stream Single Data Stream)型，只有一个进程的算法，就是传统的串行算法，适用于这类计算机．并行计算机与此不同，它每一时刻可按多条指令处理多个数据，面向并行计算机具有 2 个以上进程的算法称为**并行算法**．

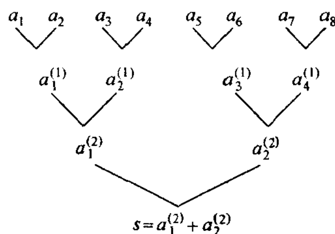
例 1　求 N 个数 $a_1, a_2, \cdots, a_n$ 的和

$$s = \sum_{i=1}^{N} a_i. \tag{1}$$

一个进程的串行算法是累加算法

$$s_1 = a_1, \quad s_k = s_{k-1} + a_k, \quad k = 2, \cdots, N.$$

$s_N = s$ 即为所求，在累加过程中所给和数规模逐次减 1，显然它不适合于并行计算．附图 1 给出 $N=8$ 时的一种并行算法，称为扇入加法．它可组成 4 个进程 P_1, P_2, P_3, P_4 如下：



附图 1 扇入加法示意图

P_1	P_2	P_3	P_4
$a_1^{(1)}=a_1+a_2$	$a_2^{(1)}=a_3+a_4$	$a_3^{(1)}=a_5+a_6$	$a_4^{(1)}=a_7+a_8$
$a_1^{(2)}=a_1^{(1)}+a_2^{(1)}$		$a_2^{(2)}=a_3^{(1)}+a_4^{(1)}$	
$s=a_1^{(2)}+a_2^{(2)}$			

在有 4 台处理机的并行系统中用 3＝lb8 级完成 8 个数求和. 第 1 级并行做 4 个加法,第 2 级并行做 2 个加法,第 3 级做 1 个加法.

并行机必须是包含 2 台以上处理机,它有不同类型. 用一个控制器控制多台处理机,在每一时刻都执行同一指令对单个或一对数组进行同样加工,这类并行机称为单指令流多数据流系统,简称 SIMD(Single Instruction Stream Multiple Data Stream)机. 另一类并行机由多个控制器分别控制多台处理机,各处理机在自己的指令控制下处理自己的数据流,称为多指令流多数据流系统,简称 MIMD(Multiple Instruction Stream Multiple Data Stream)机.

按指令流单个还是多个并行算法可分为两大类:**SIMD 算法**与 **MIMD 算法**.

SIMD 算法的基本特征是:

(1) k 个进程由 1 个指令流控制,一个并行步仅由一条指令控制,故每个并行步所含的操作必须完全相同.

(2) 允许并行步中操作的个数在 2 至 k 之间有任意性,且有含 k 个操作的并行步.

(3) 能在一个具有 k 台处理机的 SIMD 系统中实现.

例 1 给出的扇入加法就是 SIMD 算法.

MIMD 算法的基本特征是：

(1) k 个进程由 k 个指令流控制. 故每个并行步所含各操作可两两相异，而且存在包含不同操作的并行步.

(2) 同 SIMD 算法特征(2).

(3) 能在一个具有 k 台处理机的 MIMD 系统中实现.

按照进程之间是否需要同步也可将并行算法分为同步算法与异步算法.

同步算法 是指在 k 个进程的算法中有些进程的若干算法必须在另一些进程的某些算法之后执行，为此有些进程可能出现在计算之前或计算之间的等待阶段，同步算法可在一个具有 k 台处理机的 SIMD 系统或 MIMD 系统中实现.

异步算法 k 个进程间有信息联系但不须同步，它只能在一个具有 k 台处理机的 MIMD 系统中实现. 因此异步算法一定是 MIMD 算法. 使用这种算法时各次执行的实际过程可能互不重复. 同步算法既可以是 SIMD 算法也可以是 MIMD 算法，而 SIMD 算法一定是同步算法. 进程间的同步通过单指令流的控制来保证. 这样并行算法可分成 3 类，即 SIMD 算法，同步 MIMD 算法和异步算法.

例 2 对 $f(x)=0$ 求根的牛顿法(第 7 章(4.2))分别设计两个进程的同步及异步算法.

解 同步算法：可将每次迭代分成三个计算元：分别计算 $f(x_k)\to f_k, f'(x_k)\to f'_k$ ，$x_k-\dfrac{f_k}{f'_k}\to x_{k+1}$ 及检验精度. 将计算分为两个进程 P_1 及 P_2，假定计算 $f'(x_i)$ 比 $f(x_i)$ 更花时间，则两个进程或其中一个必出现等待继续计算所需数据的情况，附图 2 给出两种同步 MIMD 算法示意图.

异步算法：可引进 3 个公用变量 t_1, t_2, t_3 分别表示 $f(x)$，

$f'(x)$及 x 在计算中的当前值,仍假定计算 $f'(x)$比计算 $f(x)$更花时间,附图 3 给出两个进程的异步算法,进程 P_1 更新 t_1 与 t_3,且检验根的近似是否满足精度要求,进程 P_2 只更新 t_2,假定变量初值 $t_1=0, t_2=c\neq 0, t_3=x_0$ 前三个近似值

$$x_1 = x_0 - f(x_0)/c,$$
$$x_2 = x_1 - f(x_1)/f'(x_0),$$
$$x_3 = x_2 - f(x_2)/f'(x_1).$$

P_1	P_2
$f(x_0)\to f_0$	$f'(x_0)\to f_0'$
等待 f_0'	
$x_0-f_0/f_0'\to x_1$ 检验精度	等待 x_1
$f(x_1)\to f_1$	$f'(x_1)\to f_1'$
等待 f_1'	
$x_1-f_1/f_1'\to x_2$ 检验精度	等待 x_2

P_1	P_2
$f(x_0)\to f_0$	$f'(x_0)\to f_0'$
等待 x_1	$x_0-f_0/f_0'\to x_1$ 检验精度
$f(x_1)\to f_1$	$f'(x_1)\to f_1'$
等待 x_2	$x_1-f_1/f_1'\to x_2$ 检验精度

附图 2 MIMD 算法示意图

P_1	P_2
$f(t_3)\to t_1$	$f'(t_3)\to t_2$
$t_3-t_1/t_2\to t_3$,检验	
$f(t_3)\to t_1$	$f'(t_3)\to t_2$
$t_3-t_1/t_2\to t_3$,检验	
$f(t_3)\to t_1$	$f'(t_3)\to t_2$
$t_3-t_1/t_2\to t_3$,检验	

附图 3 异步算法示意图

其一般关系如下:

$$x_{k+1} = x_k - f(x_k)/f'(x_j), \quad k=0,1,\cdots, \quad j<k. \tag{2}$$

当 $k\to\infty$ 时 $j\to\infty$，这与传统牛顿法(4.2)不同，它是一种混乱迭代，其收敛性要另外证明，由于"计算 $f'(x)$ 比 $f(x)$ 更花时间"，所以，只要 $f'(x_j)$ 计算出新值就用于迭代，对异步算法，由于不用等待故它更节省机时.

2 并行算法基本概念

评价和分析并行算法，主要应关注它的计算复杂性，即算法的运行时间(时间复杂性)与所提供的处理机台数(空间复杂性)，并行处理的基本思想是用增加处理机台数来换取运算时间的节省，为了评价并行算法我们先引进一些基本概念.

定义 1 一个算法的**并行度**是指算法中能用一个运算步(并行)完成的运算个数. 假设算法运算个数为 r，利用 s 个运算步完成，则 r/s 称为**平均并行度**.

如例 1 的 N 个数求和(1)，若用串行算法需要 $N-1$ 个加法步，每步一个加法，并行度为 1，平均并行度 $(N-1)/(N-1)$ 也等于 1，如果用扇入加法，对 $N=2^n$ 个数求和，有 $n=\mathrm{lb}N$ 个加法步，并行性由高到低分布，逐步减半，运算个数 $r=\frac{N}{2}+\frac{N}{2^2}+\cdots+1=2^n-1=N-1$，运算步 $s=n=\mathrm{lb}N$，其平均并行度为

$$\frac{r}{s}=\frac{N-1}{\mathrm{lb}N}=O\left(\frac{N}{\mathrm{lb}N}\right). \tag{3}$$

N 个数求和的 $N-1$ 个加法不能用一个并行步完成，即不能以 $N-1$ 为并行度，但用扇入加法可把并行度降为 $\frac{N-1}{\mathrm{lb}N}$.

并行度和平均并行度是算法内在并行性的一种度量，不依赖于并行系统中处理机台数，当然处理机台数会影响完成计算所需时间.

处理机台数充分多时的最少运行时间称作算法的时间界，而

算法的运行时间达到时间界时需要提供的处理机台数称作处理机台数界，上述 N 个数求和的时间界为 $T=\mathrm{lb}N$，而处理机台数界为 $P=\dfrac{N}{2}$.

下面考察由 P 台处理机组成的并行系统，假定每台处理机的算术运算的操作时间相同，我们引进加速比和效率的概念作为并行化的又一种度量.

定义 2 设 T_1 为串行算法在单处理机上运行时间，T_P 为并行算法在 P 台处理机的系统上运行时间，则一个**并行算法的加速比**定义为

$$S_P = \frac{T_1}{T_P}, \tag{4}$$

该**算法的效率**定义为

$$E_P = \frac{S_P}{P}. \tag{5}$$

加速比和效率是评价并行算法的重要指标，研究并行算法的目标是达到尽可能大的加速比，理想上 $S_P=P$，这时并行效率达到 $E_P=1$，但一般不可能达到理想的加速比 $S_P=P$. 这是因为：

(1) 算法缺乏必要的并行度，

(2) 在并行系统上没有达到完全的负荷均衡，

(3) 通讯、存储争用及同步时间延迟等.

3 并行算法设计与二分技术

"分而治之"是并行算法设计的重要原则，其基本思想是把问题依次划分为可以独立完成的较小问题，将规模逐次减半的**二分技术**是并行算法设计的一种基本技术，例 1 给出求和的扇入加法就是一种二分技术，设 $N=2^n$ 将所给和式下标为奇偶的对应项两两合并，得

$$s = \sum_{i=1}^{N/2} (a_{2i-1} + a_{2i}).$$

这样，若令

$$a_i^{(1)} = a_{2i-1} + a_{2i}, \quad i = 1,2,\cdots,N/2,$$

则所给和式的规模被压缩了一半

$$s = \sum_{i=1}^{N/2} a_i^{(1)}.$$

重复施行这种规模减半的二分手续，二分 k 次后和式的项数被缩减成 $N/2^k$

$$s = \sum_{i=1}^{N/2^k} a_i^{(k)}.$$

式中

$$a_i^{(k)} = a_{2i-1}^{(k-1)} + a_{2i}^{(k-1)}, \quad i = 1,2,\cdots,N/2^k. \tag{6}$$

这样二分 $n=\mathrm{lb}N$ 次后，所给和式最终退化为一项，从而直接得出所求和值 s. 于是有**求和二分算法**：对 $k=1,2,\cdots,n(=\mathrm{lb}N)$ 执行算式(6)得

$$s = a_1^{(n)}.$$

附表 1 就 $N=8$ 的情形具体给出该算法计算流程.

附表 1　求和二分算法流程

<table>
<tr><td>a_1</td><td>a_2</td><td>a_3</td><td>a_4</td><td>a_5</td><td>a_6</td><td>a_7</td><td>a_8</td></tr>
<tr><td colspan="2">$a_1^{(1)}$</td><td colspan="2">$a_2^{(1)}$</td><td colspan="2">$a_3^{(1)}$</td><td colspan="2">$a_4^{(1)}$</td></tr>
<tr><td colspan="4">$a_1^{(2)}$</td><td colspan="4">$a_2^{(2)}$</td></tr>
<tr><td colspan="8">$a_1^{(3)}$</td></tr>
</table>

从以上求和二分算法看到它有以下特点：

(1) 结构递归，每步都是同样求和问题.

(2) 规模递减，和式规模逐次减半.

概括地说,二分算法的设计原理是反复地将所给计算问题加工成规模减半的同类计算问题,直至规模为 1 时直接得出问题的解.

上述求和的并行算法加速比为 $S\approx\frac{N}{\mathrm{lb}N}$,而处理机台数界为 $P=\frac{N}{2}$. 故在具有 $P=\frac{N}{2}$ 台处理机并行系统中其并行效率为

$$E \approx \frac{2}{\mathrm{lb}N}.$$

下面再讨论多项式求值问题

$$p(x) = \sum_{i=1}^{N} a_i x^{i-1}. \tag{7}$$

显然 $x=1$ 时就是求和问题(1). 故可仿照求和二分法,将所给多项式(7)的奇偶项两两合并,则有

$$\begin{aligned} p = p(x_0) &= \sum_{i=1}^{N/2} (a_{2i-1} x_0^{2i-2} + a_{2i} x_0^{2i-1}) \\ &= \sum_{i=1}^{N/2} (a_{2i-1} + a_{2i} x_0) x_0^{2(i-1)}. \end{aligned}$$

这样,若令

$$a_i^{(1)} = a_{2i-1} + a_{2i} x_0, \quad i = 1,2,\cdots,\frac{N}{2}, \quad x_1 = x_0^2,$$

则有

$$p = \sum_{i=1}^{N/2} a_i^{(1)} x_1^{i-1}.$$

这样加工得出的是一个以 x_1 为变元的多项式,而其规模(项数)是原来的一半,因此上述手续是一种二分手续. 重复这种手续,二分 k 次后所给多项式被加工成

$$p = \sum_{i=1}^{N/2^k} a_i^{(k)} x_k^{i-1} \qquad (\text{记 } a_i^{(0)} = a_i),$$

这里

$$\left.\begin{aligned} a_i^{(k)} &= a_{2i-1}^{(k-1)} + a_{2i}^{(k-1)} x_{k-1}, \quad & i &= 1,\cdots,\frac{N}{2^k}; \\ x_k &= x_{k-1}^2, & k &= 1,2,\cdots,n. \end{aligned}\right\} \tag{8}$$

这样二分 $n=\mathrm{lb}N$ 次，最终得出的系数 $a_1^{(n)}$ 即为所求的值 p.

于是得**多项式求值的二分算法**：

对 $k=1,2,\cdots,n(=\mathrm{lb}N)$ 执行算式(8)，结果有

$$p = a_1^{(n)}.$$

现在分析算法的效率，设将(8)式的 $a_i^{(k)}$ 与 x_k 并行计算(x_k 可表示为 $x_k=0+x_{k-1}\cdot x_{k-1}$ 则与 a_k 同步)，则每步 2 次运算(一乘一加)，上述算法共做 $n=\mathrm{lb}N$ 步，故其时间界为

$$T = 2\mathrm{lb}N.$$

而按(8)并行计算第 k 步需处理机 $N/2^k+1$ 台，因此处理机台数界

$$P = \max_{1\leqslant k\leqslant n}\left(\frac{N}{2^k}+1\right) = \frac{N}{2}+1.$$

注意到多项式求值的串行算法需做 $T_1=2(N-1)$ 次运算，易知此算法的加速比与效率为

$$S_P = \frac{N-1}{\mathrm{lb}N} \approx \frac{N}{\mathrm{lb}N},$$

$$E_P = \frac{2}{\mathrm{lb}N}.$$

上面分析二分法效率是在提供 $\frac{N}{2}+1$ 台处理机的系统上得到的，如果处理机只有 $P+1$ 台，而 $P<\frac{N}{2}$，若假定 $N=rP$，$r>2$ 为正整数，此时可将所给多项式(7)分成 P 段，每段含 r 项进行处理.

可利用串行算法来改造或设计并行算法. 许多传统的数值计算方法虽然只适合于串行机上计算，但其中有不少算法包含了可

直接利用的并行性，例如，上述多项式求值的秦九韶算法可用二分技术改造为适合于并行系统的并行算法. 用二分技术还可将许多矩阵计算及解线性方程组直接解法改造成适合于并行计算的算法. 还有如解线性方程组的雅可比迭代法本身就具有直接的并行性，还有一些迭代法经过重新排序也可改造成适合于并行的算法. 另一类就是根据并行算法特点设计具有新思想的新算法，它的出发点仍然是“分而治之”的原理，符合此原理的区域、算子、系统的分裂方法和技术是设计和实现并行处理的重要手段，如解线性与非线性方程的多分裂方法，解微分方程的区域分解法（即 DDM 方法），都属于此类. 由于这些方法减小了求解方程的规模，使计算时间减少，因此效率大为提高. 此外，异步数值算法基本上是混乱迭代法，是并行算法最富有特色的组成部分之一，混乱迭代不是传统意义的迭代法，在理论上必须做收敛性与收敛速度的分析，混乱迭代与多分裂技术的结合是近十几年才发展起来的新算法，仍有不少问题值得研究.

关于具体数值计算方法的并行算法可参看文献[23].

参考文献

1 李庆扬,易大义,王能超.现代数值分析.北京:高等教育出版社,1995

2 关治,陆金甫.数值分析基础.北京:高等教育出版社,1998

3 冯康等编.数值计算方法.北京:国防工业出版社,1978

4 黄友谦,李岳生.数值逼近.第2版.北京:高等教育出版社.1987

5 徐利治,王仁宏,周蕴时.函数逼近的理论与方法.上海:上海科学技术出版社,1983

6 沈燮昌.多项式最佳逼近的实现.上海:上海科学技术出版社,1984

7 Stoer J. and Bulirsch R. Introduction to Numerical Analysis. New York: Springer-Verlag, 1980

8 Wilkinson J H. Rounding Errors in Algebraic Processes. London: H. M. Stationery Office, 1963

9 Moore R E. Interval Analysis. New Jersey: Prentice-Hall, 1966

10 Nürnberger G. Approximationly Spline Functions. Berlin: Spring-Verlag, 1989

11 Nussbanmer H J. Fast Fourier Transform and Convolution Algorithms. Berlin: Spring-Verlag, 1981

12 赵访熊,李庆扬.富利叶变换滤波在地震勘探数字处理中的应用.清华大学学报,1978(4)

13 Stewart G W 著.王国荣等译.矩阵计算引论.上海:上海科学技术出版社,1980

14 Drtega J M 著.张丽君等译.数值分析.北京:高等教育出版社,1973

15 Varga R S 著.蒋尔雄译.矩阵迭代分析.上海:上海科学技术出版社,1966

16 Hageman L A, Young D M 著.蔡大用等译.实用迭代法.北京:清华大学出版社,1981

17 李庆扬,莫孜中,祁力群.非线性方程组的数值解法.北京:科学出版社,1987

18 清华大学、北京大学计算方法编写组. 计算方法. 北京:科学出版社,1974

19 Golub G H, Van Loan C F 著. 廉庆荣等译. 矩阵计算. 大连:大连理工大学出版社,1988

20 Wilkinson J H 著. 石钟慈等译. 代数特征值问题. 北京:科学出版社,1987

21 李庆扬. 常微分方程数值解法(刚性问题与边值问题). 北京:高等教育出版社,1992

22 袁兆鼎,费景高,刘德贵. 刚性常微分方程初值问题的数值解法. 北京:科学出版社,1987

23 陈景良. 并行算法引论. 北京:石油工业出版社,1992

24 李晓梅,蒋增荣. 并行算法. 长沙:湖南科学技术出版社,1992

25 李庆杨,关治,白峰杉. 数值计算原理. 北京:清华大学出版社,2000

部分习题答案

第 1 章

1. δ；2. $0.02n$；4. 1.05×10^{-3}，0.215，10^{-5}；6. $\frac{1}{2}\times10^{-3}$；9. 边长误差 $<0.005\text{cm}$；11. 不稳定，$\varepsilon^*(y_{10})\approx\frac{1}{2}\times10^{8}$；13. 0.3×10^{-2}，0.834×10^{-6}.

第 2 章

1. $5x^2/6+3x/2-7/3$；2. -0.620219，-0.616839；3. 1.06×10^{-8}；6. $h\leqslant0.006$；7. $\Delta^4 y_n=2^n$，$\delta^4 y_n=2^{n-2}$；14. 1，0；16. $P(x)=x^2(x-3)^2/4$；18. $|R_1(x)|\leqslant h^2/4$；19. $|R_4(x)|\leqslant h^4/16$.

第 3 章

1. $B_1(f,x)=x$，$B_3(f,x)=1.5x-0.402x^2-0.098x^3$；

4. (1) $\|f\|_\infty=1$，$\|f\|_1=1/4$，$\|f\|_2=1/\sqrt{7}$，(2) $\|f\|_\infty=1/2$，$\|f\|_1=1/4$，$\|f\|_2=\sqrt{\frac{5}{6}}$，(3) $\|f\|_\infty=\left(\frac{1}{2}\right)^{m+n}$，$\|f\|_1=\frac{n!\ m!}{(n+m+1)!}$，$\|f\|_2=\sqrt{\frac{(2n)!\ (2m)!}{[2(n+m)+1]!}}$，(4) $\|f\|_\infty=4/\text{e}$，$\|f\|_1=5-10/\text{e}$，$\|f\|_2=7(3/4-4/\text{e}^2)$；

6. (1) 不构成内积，(2) 构成内积；

7. $T_0^*(x)=1$，$T_1^*(x)=2x-1$，$T_2^*(x)=8x^2-8x+1$，$T_3^*(x)=32x^3-48x^2+18x-1$；

8. $\varphi_0(x)=1$，$\varphi_1(x)=x$，$\varphi_2(x)=x^2-\frac{2}{5}$，$\varphi_3(x)=x^3-\frac{9}{14}x$；

11. $P_0(x)=(M+m)/2$；12. 3/4；13. $P_1(x)=\frac{2}{\pi}x+0.105257$；

14. $P_1(x)=(\mathrm{e}-1)x+\frac{1}{2}[\mathrm{e}-(\mathrm{e}-1)\ln(\mathrm{e}-1)]$；15. $P_3^*(x)=5x^3-\frac{5}{4}x^2+\frac{1}{4}x-\frac{129}{128}$；16. $S^*(x)=0.1171875+1.640625x^2-0.8203125x^4$；

17. (1) $S_1^*(x)=-0.2958x+1.1410$，(2) $S_1^*(x)=0.1878x+1.6244$，(3) $S_1^*(x)=-0.24317x+1.2159$，(4) $S_1^*(x)=0.6822x-0.6371$；

18. $S_3^*(x)=1.5531913x-0.5622285x^3$；19. $S(t)=22.25376t-7.855048$；20. $y(x)=0.9726046+0.0500351x^2$；

21. $y=5.2151048\mathrm{e}^{-\frac{7.4961692}{t}}$；23. $R_{22}(x)=3-\frac{4}{x+0.5}+\frac{1.25}{x+1.5}$；

24. $R_{33}(x)=\frac{60x-7x^3}{60+3x^2}$；25. $R_{21}(x)=\frac{6+4x+x^2}{6-2x}$.

第 4 章

1. 1) $A_{-1}=A_1=h/3, A_0=4h/3$，具有 3 次代数精度，2) $A_{-1}=A_1=8h/3$，$A_0=-4h/3$，3 次代数精度，3) $x_1=-0.28990$，$x_2=0.62660$ 或 $x_1=0.68990$，$x_2=-0.12660$，2 次代数精度，4) $\alpha=1/12$，3 次代数精度；2. 1) $T_8=0.11140$，$S_4=0.11157$，2) $T_{10}=1.39148$，$S_5=1.45471$，3) $T_4=17.22774$，$S_2=17.32222$，4) $T_6=1.03562$，$S_3=1.03577$；4. $S_1=0.63233$，误差 0.00035；

6. $n\geqslant 213$，用辛普森公式区间分为 8 等分；8. (1) 0.71327，(2) 0，(3) 10.1517434；9. $n=2$，$I\approx 10.9484$，$n=3$，$I\approx 10.95014$，(准确值 $I=10.9517032$)；10. 48708 km；11. 3.14158； 12. 1) 1.09863，2) 1.08940，1.09862，3) 1.09854；13. 用三点公式求得一阶导数值 −0.247，−0.217，−0.189，用五点公式得 −0.2483，−0.2163，−0.1883.

第 5 章

4.

$$\begin{cases} l_{i1} = a_{i1} \quad (i = 1,2,\cdots,n), \\ u_{1j} = a_{1j}/l_{11} \quad (j = 2,3,\cdots,n), \\ l_{ik} = a_{ik} - \sum_{r=1}^{k-1} l_{ir}u_{rk} \quad (i = k,\cdots,n), \\ u_{kj} = \left(a_{kj} - \sum_{r=1}^{k-1} l_{kr}u_{rj}\right)/l_{kk} \quad (j = k+1,\cdots,n); \end{cases}$$

5. (a) 设 $\boldsymbol{U}$ 为上三角阵

$$x_n = b_n/u_{nn},$$

$$x_i = \left(b_i - \sum_{j=i+1}^{n} u_{ij}x_j\right)/u_{ii} \quad (i = n-1, n-2, \cdots, 1),$$

(b) $n(n+1)/2$,

(c) 记 $\boldsymbol{U}^{-1}$ 的元素为 s_{ij}，$\boldsymbol{U}$ 的元素记为 u_{ij}：

$$\begin{cases} s_{ii} = 1/u_{ii} (i = 1,2,\cdots,n), \\ s_{ij} = -\sum_{k=i+1}^{j} u_{ik}s_{kj}/u_{ii}, \\ \quad i = n-1, n-2, \cdots, 1, j = i+1, \cdots, n; \end{cases}$$

7.

$$\boldsymbol{A}^{-1} = \begin{pmatrix} -0.04705885 & 0.5882353 & -0.2705882 & -0.9411764 \\ 0.3882353 & -0.3529412 & 0.4823529 & 0.7647058 \\ -0.2235294 & 0.2941177 & -0.03529412 & -0.4705882 \\ -0.03529412 & -0.05882353 & 0.04705882 & 0.2941176 \end{pmatrix}$$

8. (1) $\beta_1 = -1/2$, $\beta_2 = -2/3$, $\beta_3 = -3/4$, $\beta_4 = -4/5$, (2) 解 $\boldsymbol{Ly} = \boldsymbol{f}$, $\boldsymbol{y} = (1/2, 1/3, 1/4, 1/5, 1/6)^{\mathrm{T}}$, (3) 解 $\boldsymbol{Ux} = \boldsymbol{y}$, $\boldsymbol{x} = (5/6, 2/3, 1/2, 1/3, 1/6)^{\mathrm{T}}$；

9. $\boldsymbol{x} = (1.11111, 0.77778, 2.55556)^{\mathrm{T}}$；

10. (a) $\boldsymbol{A}$ 不能分解为三角阵的乘积，但换行后可以.(b) $\boldsymbol{B}$ 可以但不唯一，$\boldsymbol{C}$ 可以且唯一；

11. $\|\boldsymbol{A}\|_\infty = 1.1$, $\|\boldsymbol{A}\|_1 = 0.8$, $\|\boldsymbol{A}\|_2 = 0.825$, $\|\boldsymbol{A}\|_F = 0.8426$；

18. $\mathrm{cond}(\boldsymbol{A})_\infty = 39601$, $\mathrm{cond}(\boldsymbol{A})_2 = 39206$.

第 6 章

1. (a) 两种方法均收敛,(b) 用雅可比迭代法迭代 18 次,

$x^{(18)}=(-3.9999964, 2.9999739, 1.9999999)^T$,用高斯-塞德尔迭代法迭代 8 次,

$x^{(8)}=(-4.000036, 2.999985, 2.000003)^T$;

2. (a) 雅可比迭代法不收敛,高斯-塞德尔迭代法收敛,

(b) 雅可比迭代法收敛,高斯-塞德尔迭代法不收敛;

5. $\omega=1.03$ 时迭代 5 次达到精度要求 $x^{(5)}=(0.5000043, 0.1000001, -0.4999999)^T$,$\omega=1$ 时迭代 6 次达到精度要求 $x^{(6)}=(0.5000038, 0.1000002, -0.4999995)^T$,$\omega=1.1$ 时迭代 6 次达到精度要求,$x^{(6)}=(0.5000035, 0.9999989, -0.5000003)^T$;

6. $\omega=0.9$,迭代 8 次时达到精度要求,

$x^{(8)}=(-4.000027, 0.2999989, 0.2000003)^T$;

第 7 章

1. 0.512;

2. 1)和 2)收敛,3)发散;

3. 1)二分 14 次得 0.0905456,2)迭代 5 次得 0.0905264;

5. (2) $x_0=1.5$, $x_2=1.465572$, (3) $x_0=1.5$, $x_3=1.465571$;

7. (1) $x_3=1.8794$, (2) $x_3=1.8794$; 8. 4.49342;

11. 牛顿法 $x_{20}=1.895494$.其他方法迭代次数 $n=4,13,10,723805$;

14. $-(n-1)/2\sqrt[n]{a}$,$(n+1)/2\sqrt[n]{a}$;15. $1/4a$;

16. $(1.526824, 0.508139)^T$.

第 8 章

1. (a) 取 $v_0=(1,1,1)^T$, $\lambda_1\approx 9.6058$, $x_1\approx(1, 0.6056, -0.3945)^T$, (b) 取 $v_0=(1,1,1)^T$, $\lambda_1\approx 8.86951$, $x_1\approx(-0.60422, 1, 0.15094)^T$; 2. $\lambda=7.288$, $x\approx(1, 0.5229, 0.2422)^T$;

5. $\boldsymbol{u}_1=\begin{pmatrix}1 & 0 & 0\\ 0 & -\frac{3}{5} & -\frac{4}{5}\\ 0 & -\frac{4}{5} & \frac{3}{5}\end{pmatrix}$，$\boldsymbol{A}_2=\begin{pmatrix}1 & -5 & 0\\ -5 & \frac{77}{25} & \frac{14}{25}\\ 0 & \frac{14}{25} & -\frac{23}{25}\end{pmatrix}$；

7. (a) $\boldsymbol{A}$ 的特征值为 $\lambda_1=\frac{1}{2}+\frac{\sqrt{33}}{2}$，$\lambda_2=2$，$\lambda_3=\frac{1}{2}-\frac{\sqrt{33}}{2}$，

(b) $\boldsymbol{B}$ 的特征值为 $\lambda_1=2+\sqrt{3}$，$\lambda_2=2$，$\lambda_3=2-\sqrt{3}$.

选取位移 $s_k=b_{33}^{(k)}$，

$$\boldsymbol{B}_S=\begin{pmatrix}3.7316925974 & 0.0249060210 & 0.0\\ 0.0249060210 & 2.0003582102 & \varepsilon\\ 0.0 & \varepsilon & 0.2679491924\end{pmatrix}$$

其中 $|\varepsilon|<5\times10^{-11}$；

8. $\boldsymbol{Q}=\frac{1}{3}\begin{pmatrix}1 & 2 & 2\\ 2 & 1 & -2\\ 2 & -2 & 1\end{pmatrix}$，$\boldsymbol{R}=\begin{pmatrix}3 & -3 & 3\\ & 3 & -3\\ & & 3\end{pmatrix}$；

9. $\boldsymbol{A}$ 的特征值 $2\leqslant\lambda\leqslant6$，$\boldsymbol{A}^{-1}$ 特征值 $\frac{1}{6}\leqslant\mu\leqslant\frac{1}{2}$.

第 9 章

1. 0，0.0010，0.0050；2. 0.145；4. 0.500，1.142，2.501，7.245；

5. 1) 1.2428，1.5836，2.0442，2.6510，3.4365，

   2) 1.7276，2.7430，4.0942，5.8292，7.9960；

8. (1) $0<h\leqslant0.02$，(2) $0<h\leqslant0.0278$，(3) $0<h<+\infty$；

9. 显式 0.626，隐式 0.633，真值 0.6321；

10. $-\frac{5}{8}h^3y'''(x_n)$；12. 刚性比 $s=20$，$0<h<0.139$.